DELFT UNIVERSITY OF TECHNOLOGY

MASTER THESIS

Switching from Open Source to Cloud Protection License: What is the impact on Community Health?

Joel Koch

Student ID: 5242347

March 21, 2023

Delft University of Technology

Switching from Open Source to Cloud Protection License: What is the impact on Community Health?

Master thesis submitted to Delft University of Technology

in partial fulfilment of the requirements for the degree of

MASTER OF SCIENCE

in **Management of Technology**

Faculty of Technology, Policy and Management

by

Joel Koch

Student number: 5242347

To be defended in public on 04-04-2023

Graduation committee

Chairperson	: Dr. ir., G.A, de Reuver, ICT
First Supervisor	: Dr. ir., J., de Stefani, ICT
Second Supervisor	: Dr., G., van de Kaa, ETI

Executive Summary

Open source software is created in a collaborative approach in public communities, using open source licenses. This thesis investigates the phenomenon of companies which developed open source software but adopted cloud protection licenses, which represent a novel type of software licenses. These cloud protection licenses protect their business from competitors, by preventing the latter to copy their products. At the same time, the change of license potentially affects the community and therefore their previous approach of software development.

Correspondingly, this thesis addresses the practical problem of executives of the companies who miss information about the effects of the license change on the health of their community. From a scientific perspective, there is little knowledge about the relationship between license and community in the specific context of commercial open source software. Furthermore, there is no existing research about cloud protection licenses. This leads us to the main research question: *What is the impact on community health when a commercial open source software company changes its licensing terms from an open source license to a cloud protection license?*

The thesis presents the required background information about the context of the research. Various software licenses are introduced, with a focus on open source licenses and cloud protection licenses. Furthermore, background on the commercialization of open source software is presented, followed by an explanation of the relation between open source licenses and a software delivery model, software as a service. The concept of open source community health is introduced which the thesis defines as the part of open source project health with focuses on the community. In particular, we consider community activity and community structure as indicators of community health. Moreover, terms specific to software development that are used in the thesis are described.

In the following, an overview of the research around open source projects is presented. In more detail, the thesis describes the research investigating community activity, community structure, and license choice and license change.

Based on this research, five propositions about the impact of the license change on community activity and community structure are presented. First, changing the licensing terms from an open source license to a cloud protection license leads to reduced community activity. Second, changing the license from an open source license to a cloud protection license leads to less individuals joining the community. Third, changing the license from an open source license to a cloud protection license leads to more community members leaving the community. Fourth, changing the license from an open source license to a cloud protection license leads to increased knowledge concentration among individuals in the community. Fifth, changing the license from an open source license to a cloud protection license leads to increased knowledge concentration considering organizations in the community.

To examine these propositions with data from real software projects, we use a methodology which combines an existing tool, CHAOSS, with Python scripts to visualize the data. CHAOSS retrieves data from the Git repositories and GitHub projects and enriches the data with information about the background of community members. The Python scripts extract the data from CHAOSS, calculate the metrics described in the following paragraph, and visualize the results. We evaluate the

propositions through a visual inspection of the resulting plots around and after the dates of license change.

To examine the first proposition, both major types of community activity are evaluated, technical activity and social activity. Considering technical activity, the development of the number of monthly commits is investigated, which is extracted from the respective Git repository. In respect to social activity, the development of the number of monthly created issues is examined, which is retrieved from the respective GitHub project. For both types of activity, the time series data is decomposed into seasonal, trend, and residual component to facilitate the analysis. To examine the second proposition, the number of monthly joining members is calculated from the date of the first contribution by each individual member stored in Git. Similarly, the number of monthly leaving members is calculated from the date of the last contribution by each individual member stored in Git to evaluate the third proposition. To evaluate the knowledge concentration among community members, the onion model is applied which associates each member with a role for each quarter. Core members are responsible for 80% of all contributions in the quarter, regular members for 15%, and casual members for the remaining 5%. To examine the last proposition, the proportion of contributions authored by employees of the respective company is determined for each month and the development evaluated.

This methodology is applied to projects of MongoDB, Elasticsearch and Redis. In the data from the three companies we could not find support for the first proposition, a reduction in community activity, but contradicting developments in all projects. Regarding the proposition of a decrease in joining members, the decomposed time series data indicates contradicting developments in the projects of MongoDB and Redis. When assessing the proposition of an increase of leaving members, we found that the decomposed time series data is congruent with the proposition in the project of MongoDB. However, the data of the projects of Elasticsearch and Redis indicates no support for it. To evaluate the proposition of an increased knowledge concentration among community members, the development of the shares of the onion roles within the community is analyzed, and the results contradict the proposition. Considering the proposition of an increased knowledge concentration among organizations in the community, there is no contradicting evidence in the data and the developments in the projects are congruent with the proposition.

In summary, the data indicates that the impact of a license change to a cloud protection license on community health is rather small and constrained to the concentration of knowledge in respect to organizations.

Regarding the practical problem, this thesis provides executives of the companies with these insights on the potential effects of the adoption of a cloud protection license on their community. As those effects are small, we can recommend executives to adopt a cloud protection license if it addresses the strategic needs of their companies. However, they must consider the specific context of their company which might imply additional concerns. Furthermore, the thesis contributed to the research in the field of open source software, mainly by investigating cloud protection licenses, which represent a new category of licenses and were correspondingly not considered by previous research. This way, it continues the research on the impact of choice and change of software licenses on open source communities. In particular, the thesis examined the impact of a change to this new category of licenses on community activity and community structure. Moreover, it includes the distribution of roles according to the onion model and the share of contributions authored by employees of the company in the evaluation of the impact of the license change. These are two previously disregarded aspects of the structure of open source communities in the research about license choice and license change.

This thesis relates to the Management of Technology study program by taking the perspective of companies to analyze the effects of their strategic choices on their open innovation processes. By studying the effects of a license change, it examines a problem that is located at the intersection of technology, organizations, and strategy.

Several limitations apply. First, the underlying theoretical concepts were selected from the literature in the area of open source software, it could be that the transfer to commercial open source software might not be perfect. Second, there are limitations related to the specific projects. Regarding Elasticsearch, the period after the license change is relatively short. This became clear especially when the analyzed data represents quarters, where only seven data points follow the event of the license change. Considering the projects of Redis, the absolute values are simply small. Third, there are limitations related to the functioning of CHAOSS on which the results depend. The identification of community members might not work perfectly in some cases. Furthermore, there were small differences between the data extracted from CHAOSS, and the data directly presented by CHAOSS. For instance, the values of monthly joining members differed by 1 in rare cases. However, these rare occurrences do not affect the overall outcomes.

Contents

List of Figures

List of Tables

List of Abbreviations

AGPLv3 GNU Affero General Public License version 3

CHAOSS Community Health Analytics in Open Source Software

COSS commercial open source software

ELv2 Elastic License 2.0

GPL GNU General Public License

OSI Open Source Initiative

OSS community open source software

RSAL Redis Source Available License

SaaS software as a service

SSPL Server Side Public License

STL seasonal-trend decomposition based on loess

1 Introduction

Open source software is created in a collaborative approach in public communities, and it is furthermore widely known for its free availability (Colazo & Fang, 2009; Nakakoji et al., 2002; Riehle, 2019, 2020). It may be surprising but in spite of the free availability there are companies which successfully commercialize open source software (Hauge & Ziemer, 2009; Popp, 2012; Riehle, 2019, 2020, 2021).

This thesis investigates a recent phenomenon of companies which developed open source software but made the strategic decision to move away from it by adopting a novel type of software license. In this context, the thesis aims to research whether this strategic move impacted the respective open source community.

This chapter first provides a brief summary of the domain background to introduce important concepts required to follow the rest of the introduction. This includes software licenses and how open source software relates to them, as well as how companies realize commercialization of open source software. For a better understanding, a more detailed description of the domain background is given in chapter 2.

The chapter proceeds by introducing the practical and scientific problem addressed in the thesis, followed by the corresponding research questions. The final paragraph outlines the structure of the thesis.

1.1 Summary of Domain Background

When releasing software, the owners usually include its licensing terms. This license determines the set of rights and conditions that users of the software must comply with (Wilson, 2013). We can distinguish several categories of licenses according to the rights they grant (see chapter 2.1).

One of these categories are open source licenses which were specifically created to ensure "free" development of software (Riehle, 2019). In short, open source licenses grant everyone the freedom to use, redistribute, and modify the software (Riehle, 2019). This necessarily includes the public availability of the source code of the software, next to other requirements (Open Source Initiative, 2007). Software released under an open source license is called open source software (Stewart et al., 2006). As a consequence of the requirements of the license, open source software is typically developed in an open community using a collaborative approach (Nakakoji et al., 2002).

Following other researchers (Hauge & Ziemer, 2009; Popp, 2012; Riehle, 2020), open source software which is primarily created to monetize it will be called commercial open source software (COSS) in this thesis. This is in contrast to traditional community open source software (OSS), where commercialization is not a driving force.

As the source code of open source software must be publicly available for free, COSS companies usually cannot simply sell the software (Riehle, 2019). Instead, various approaches emerged to monetize COSS (see chapter 2.2).

The approach of particular importance for this thesis is the single-vendor open source business model as described by Riehle (2012) and Riehle (2020). In this model, there is a single company which controls the open source project (see chapter 2.2). The company makes all major decisions and holds the copyright of the complete source code. As a consequence, the company can decide to change the licensing terms of the software.

Another important concept in the context of the thesis is software as a service (SaaS). SaaS describes software that is delivered on demand and over the internet (Armbrust et al., 2010).

Many COSS companies implemented a SaaS model in recent years, just as regular software companies which do not develop open source software (Riehle, 2020). The main product of MongoDB, for instance, is "MongoDB Atlas", a database as a service. However, the legally granted rights of open source software, make COSS companies which offer a SaaS product vulnerable to competitors, who can simply copy their products and offer the same service (MongoDB, Inc., 2022).

Figure 1.1 illustrates this dilemma. An originator company collaborates with external contributors in the open source community to develop a software product which is provided as a service to customers. A competing company can copy the software and provide the same service to its customers, without bearing the costs of development.

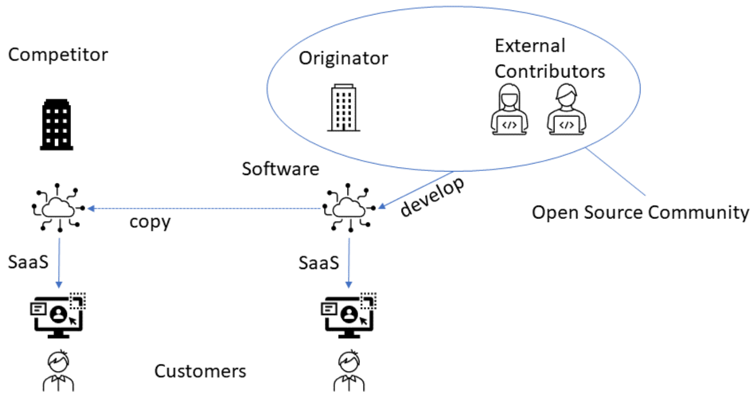

Figure 1.1: Diagram summarizing the competitive context of COSS companies with a SaaS offering. The COSS company develops the software in collaboration with the community and provides the service to its customers. A competitor copies the SaaS product to offer the same service to other customers

As a response, single-vendor COSS companies created and use new licenses which in fact prevent competing companies from copying their service. There is no existing term for this category of licenses, therefore they are called "cloud protection licenses" in this thesis

(see chapter 2.1.2).
Cloud protection licenses are not open source licenses because of the legal conditions they determine (Open Source Initiative, 2021). When adopting a cloud protection license, a company therefore stops developing open source software. Nevertheless, those companies which decided to use a cloud protection license continue to develop their software collaboratively to maintain the benefits of open source software (MongoDB, Inc., 2022). Chapter 2.3 provides a more detailed explanation of the background of cloud protection licenses.
Open source software is typically developed in a community. It is of fundamental importance for the project to maintain the community in a "healthy" state in order to ensure ongoing development. However, community health is not a well-defined concept. As we will see in chapter 2.4, researchers also use terms such as "success" or "sustainability" to describe community health. Moreover, open source communities are complex social-technological constructs. Consequently, different technological and social components contribute to the health of a community. For example, a community could create technologically perfect software but at the same time maintain a social atmosphere which is so demotivating to community members that the community cannot be described as healthy. On the other hand, the community could maintain a very positive atmosphere but develop software of low quality which will ultimately lead to the abandonment of the project.
This demonstrates the variety of aspects of community health. To find a scope suitable for a thesis, we first evaluate which perspectives exist (see chapter 2.4) and which information is accessible (see chapter 5). As a result of these considerations, this thesis uses a combination of community activity and community structure as indicators for community health. Chapter 3 provides more details about related existing research.

1.2 Practical Problem

Executives of COSS companies could decide to change the license of their software to a cloud protection license for future versions. By doing so, they stop their competitors from copying and offering their product as a service. At the same time, they will lose the "open source" label for their project.
Previous research has shown that the choice of license impacts the open source community (Colazo & Fang, 2009; Viseur & Robles, 2015). Therefore, the change of license could have negative effects on the health of the open source communities. As the open source community is an essential part of any open source business model, executives should be well-informed before making decisions which influence the community.

1.3 Scientific Problem

Researchers from various fields have investigated open source software (see chapter 3). The overwhelming majority of researchers examined traditional OSS. Correspondingly,

there is a knowledge gap around the applicability of the developed theories in the special context of COSS. For instance, the effects of the choice of license on OSS communities have been studied by Colazo and Fang (2009), Santos (2017), Stewart et al. (2006), Vendome et al. (2017), and Viseur and Robles (2015), but never with a focus on COSS.
Moreover, this thesis addresses a knowledge gap in respect to the new phenomenon of cloud protection licenses. When examining the relationship between licenses and properties of the communities, researchers studied the different open source licenses, as introduced in chapter 2.1. However, cloud protection licenses were created only recently and therefore there is no research about them yet.
Consequently, the specific knowledge gap aimed to be filled by this thesis is the impact of a license change from an open source license to a cloud protection license on the health of a COSS community.

1.4 Research Questions

The strategic move by COSS companies to switch to a cloud protection license was the starting point for this thesis. Correspondingly, this decision and its impact on the community are central to the main research question:

> *What is the impact on community health when a COSS company changes its licensing terms from an open source license to a cloud protection license?*

Before being able to answer the main research question, several steps are required. First, it is important to know how to analyze community health and which aspects to include. Considerations regarding which aspects of community health to examine are given in chapter 2.4. A suitable methodology, including the selection of tools and data sources, is presented in chapter 5
With this preparations we can proceed to the core of the thesis. To align the academic work in this thesis with the work of other researchers, existing theories are applied to propose the expected impact on activity and structure of the community. These expectations are then examined in selected projects. Correspondingly, these are the subquestions that will be answered:

1. According to literature, how could the switch to a cloud protection license impact the activity of the community?
2. According to literature, how could the switch to a cloud protection license impact the structure of the community?
3. How did the license switch impact community activity in the selected projects?
4. How did the license switch impact the structure of the community in the selected projects?

1.5 Thesis Outline

The remainder of this thesis is structured as follows: Chapter 2 provides a more detailed explanation of the domain background briefly introduced in this chapter. Next, chapter 3 describes existing research in the field of open source software. Chapter 4 builds on the previous chapter to develop propositions about which impact on community health could be expected. Subsequently, chapter 5 describes the methodology applied to analyze the projects which are introduced in chapter 6. In chapter 7 the results of the analysis are presented. Chapter 8 concludes the thesis.

2 Domain Background

This chapter presents information required to understand the background of the thesis. It begins by introducing software licenses and how they can be categorized. Open source licenses and cloud protection licenses are described in more detail. Subsequently, background on COSS is presented, followed by an explanation of the relation between SaaS and open source licenses. The concept of open source community health is introduced. Last, the chapter describes terms specific to software development that are used in the thesis.

2.1 Software Licenses

After writing the source code of a software component, in principle the author of the source code holds its copyright. Without further measures, only the copyright holder is entitled to copy, distribute or modify the software (Wilson, 2013). If the software is created by an employee of a company, the copyright will usually be transferred to the company. In order to allow other parties than the copyright holder to use the software, it must be licensed to set the legal framework (Wilson, 2013).

Software licenses can be categorized according to the rights they grant to other parties (Colazo & Fang, 2009; Hauge & Ziemer, 2009; Santos, 2017; Stewart et al., 2006; Viseur & Robles, 2015). On a high level, we can distinguish open source licenses and proprietary licenses. Open source licenses are fundamental to this work and will be explained in more detail in the following paragraph. Apart from few exceptions, those licenses which are not open source licenses form the category of proprietary licenses. Open source licenses have a common set of characteristics. Proprietary licenses on the other hand are totally free in the definition of their legal framework. For each category of software licenses introduced in this chapter, Table 1 shows whether it represents open source licenses, whether it requires the source code of the licensed software to be public, whether it allows commercial redistribution, and whether modifications are allowed.

2.1.1 Open Source Licenses

The term "Open Source Software" was introduced in the late 1990s (Open Source Initiative, 2018) to define software which license grants everyone the freedom to use, redistribute, and modify it.

In order to institutionalize these principals, the Open Source Initiative (OSI) was established in 1998 (Open Source Initiative, 2018). The OSI is the official guard of the open source idea and maintains control by examining licenses. It is broadly accepted that the OSI must approve a license before it can be called "open source" (Elasticsearch B.V., 2021; MongoDB, Inc., 2022; Timescale, Inc., 2020). Only those licenses which comply with the ten criteria of the official open source definition (Open Source Initiative, 2007) will be approved.

The most known criterion guarantees the public availability of the source code of the software. It is important that this is only one of the ten criteria. Criterion 6 is in particular relevant for the background of this thesis. It states that open source licenses must not prevent the application of the software "in a specific field of endeavor" (Open Source Initiative, 2007). A notable result of the definition of open source software is the fact that it guarantees the right for everyone to commercialize the software. Criterion 6 and the right to commercialize open source software will play a role in chapter 2.3.

Commonly, open source licenses are categorized either as copyleft or permissive licenses (Colazo & Fang, 2009; Open Source Initiative, 2022a; Stewart et al., 2006; Viseur & Robles, 2015). Copyleft licenses require everyone who modifies and redistributes the licensed software to use the original copyleft license for all of the resulting software (Open Source Initiative, 2022a). This way, copyleft licenses ensure that all modified versions will remain open source software, and improvements can be transferred back to the original project. The GNU General Public License (GPL) (Free Software Foundation, 2007) is a widely used copyleft license. If anyone modifies software which was published using the GPL, the modified software can only be distributed using the exact same GPL.

Permissive licenses on the other hand do not include such requirements. Therefore, software which is derived work of a permissively licensed open source software can be licensed in any way, including proprietary licenses. The MIT license (Open Source Initiative, 2022b) is a commonly used permissive open source license. Hofmann et al. (2013) and Vendome et al. (2017) observed a trend towards permissive license since the early 2000s. Hofmann et al. (2013) suggest that this is because of the increased amount of sponsored open source software. The companies sponsoring the projects want to be able to modify the software and use proprietary licenses for these modifications. Consequently, they prefer permissive over copyleft licenses (Hofmann et al., 2013).

2.1.2 Cloud Protection Licenses

There is one specific type of proprietary license which is central in this research. Licenses of this type share most characteristics with open source licenses, such as the publicly available source code, but introduce an additional limitation which prevents other parties from offering the licensed software as a service.

They represent a relatively new phenomenon and therefore there is no existing research on this specific category of licenses and consequently no academic term yet. The OSI referred to them as "fauxpen" source (Open Source Initiative, 2021), while for instance the blogger Igor Kotua (2022) named them "strong OSS licenses". However, both of these names do not capture the essential characteristic of preventing competitors from offering copied SaaS products. Both names are generic and could be applied to other licenses with completely different regulations.

One adopting company, TimescaleDB, uses the term "cloud protection licenses" in a blog entry (Timescale, Inc., 2020). This captures the nature of the category of licenses and emphasizes that that they prevent competing SaaS offerings. Therefore, the term used in this thesis for this category of licenses is also "cloud protection license".

The licenses which fall into this category differ from each other in the actual legal requirements they specify. As described in the next paragraph, some are based on open source licenses, and therefore inherit the requirements of the respective open source license. Furthermore, they use different mechanisms to prevent SaaS competition. The following paragraph introduces examples of cloud protection licenses to show the mechanisms.

The Server Side Public License (SSPL) (MongoDB, Inc., 2018) is based on version 3 of the GPL (MongoDB, Inc., 2022). As described above, the GPL is a copyleft license. Consequently, the SSPL inherits the regulations specified by the GPL. The additional limitation in comparison to the GPL can be found in section 13 of the license text (MongoDB, Inc., 2018). The SSPL does not directly forbid SaaS offerings but instead requires everyone who wants to offer the licensed software as a public service to publish the source code of all software required to create this service in a way that anyone could recreate the service (MongoDB, Inc., 2022). This de-facto prevents smaller competitors from creating copied SaaS offerings because they do not even have access to all the software required to create such a service and instead would rely on products of public cloud providers. Those public cloud providers could technically create a competing SaaS offering because they own the infrastructure and software required to create the service and therefore could publish the source code. Up to today there is no public cloud provider acting this way, and it will most probably remain so given that the software required to create such a service is the core of the business of a public cloud provider.

The Redis Source Available License (RSAL) grants everyone the right "to use, copy, distribute, make available, and prepare derivative works of the software" (Redis Ltd., 2022, second paragraph). At the same time it forbids making the software available to third parties (Redis Ltd., 2022). Consequently, the RSAL directly prevents competing SaaS offerings.

Another approach is the "Commons Clause" created by FOSSA Inc. (2018). The Commons Clause is a short paragraph of text that can be combined with any open source license. It determines that there is no "right to Sell the Software" (FOSSA Inc., 2018). Then, it defines selling in a very broad way, including and therefore directly prohibiting SaaS offerings.

Chapter 2.3 provides more details about the background of cloud protection licenses.

2.2 Commercial Open Source Software

Per definition, open source software must be available for usage at no cost. Nevertheless, companies can monetize open source software.

Even before the establishment of the OSI, and therefore before the label "open source" officially existed, different models emerged to form businesses around open source software. Often, their business models are on first impression similar to those of proprietary software vendors. However, the free and open availability of the product has implications

License	Copyleft	Permissive	Cloud Protection	Proprietary
Open Source	yes	yes	no	no
Public Source Code	yes	yes	yes	usually no, but possible
Commercial Redistribution	yes	yes	yes, but not as a service	usually no
Modifications allowed	yes, but must use same copyleft license	yes	yes, but must use same license	usually no
Examples	GPL	MIT License	SSPL	

Table 1: Properties of categories of software licenses

on many parts of the business (Hauge & Ziemer, 2009; Riehle, 2020, 2021). Most obviously, the revenue model cannot simply consist of selling the software product but must include other aspects. Nevertheless COSS companies can create revenues from intellectual property, e.g., by providing sophisticated documentation (Popp, 2012; Weikert & Riehle, 2013).

Hauge and Ziemer (2009), Popp (2012), Riehle (2012) and Riehle (2020, 2021), and others categorized open source business models in slightly different ways using different names. In the following paragraphs, the terms and categories used by Riehle are adopted. Commercial distributors (Riehle, 2021), for example of Linux operating systems, integrate countless open source projects to offer high-quality products. Companies operating according to the service and support model (Riehle, 2019) offer accompanying services to users, such as support or hosting of the software (Weikert & Riehle, 2013).

Of particular importance for this thesis is the single-vendor open source business model. In this model, a single company owns and manages the development of an open source software project in order to commercialize it (Riehle, 2012; Riehle, 2020).

Riehle describes the single-vendor open source business model as follows. To stay in control, contributions from external developers are only accepted if the contributor transfers the copyright or relicensing rights to the COSS company. The major part of product development is performed by employees of the vendor. Nevertheless, the open source software benefits from community members who contribute and report defaults. Single-vendor COSS companies leverage the community to recruit new employees who are already familiar with their product and proved skillful in working with it. Applying frequent feedback from the large user base, the company can prioritize features in a very efficient and targeted way. Marketing is massively improved by word of mouth marketing of happy users and their experience sharing. As customers can use the free version as long as they want, they adopt the product faster. The large scale adoption on the other hand

leads to fast detection of defaults, which are then often reported and fixed. Together, this leads to products of high quality. Sales teams can concentrate on converting free users who are already familiar with the product into paying users. Furthermore, there is minimal friction if at all to convert from free usage to a paid user. In addition, documentation and support material of high quality is often created or co-created by the community.
As described, it is essential for a COSS business to create and maintain a community to benefit from an open source strategy (Hauge & Ziemer, 2009). This includes creating official infrastructure for communication, organizing meetings and establishing relationships between employees and community users (Hauge & Ziemer, 2009). While this is an additional challenge and causes costs, in a working COSS business model the benefits outweigh the costs.

2.3 Open Source Licenses and Software as a Service

SaaS describes software that is delivered on demand and over the internet (Armbrust et al., 2010). This concept became increasingly popular with the emergence of cloud computing. COSS companies could profit tremendously from offering their software as a service in recent years. Notably firms of the third generation of single-vendor COSS companies, as described by Riehle (2020), implemented successful business models.
As most open source licenses were created before the spread of SaaS, they do not cover it in their terms (Open Source Initiative, 2022a). This is in particular relevant with respect to copyleft licenses. While these require redistributors to use the original copyleft license, it is controversial whether offering a modified version of the software as a service is redistribution in the sense of the licensing terms (Elasticsearch B.V., 2021; MongoDB, Inc., 2022). This is not the intention of the users of copyleft licenses, who want to ensure that any software derived from their work will remain open source (Open Source Initiative, 2022a). Attempts were made to regulate SaaS in new copyleft open source licenses, such as the GNU Affero General Public License version 3 (AGPLv3) (Free Software Foundation, 2007). However, there were still uncertainties around legal requirements, in particular the definition of redistribution and whether SaaS counts as such (MongoDB, Inc., 2022).
As a consequence, COSS companies who offer SaaS faced competition by companies which copy their software to create a similar SaaS product (Elasticsearch B.V., 2021; Igor Kotua, 2022; MongoDB, Inc., 2022; Timescale, Inc., 2020). This way, these competitors could profit from open source innovations of the COSS companies while this was not the case the other way around. Above all, public cloud providers (Google, Amazon, etc.) made use of their large customer base and resources to capture a significant market share by offering well-integrated copies of products of COSS companies on their platforms (Elasticsearch B.V., 2021; Igor Kotua, 2022; MongoDB, Inc., 2022; Timescale, Inc., 2020). This is in principle legally allowed, as they have the right to commercialize software with open source licenses.
As a response, COSS companies created cloud protection licenses, as described in

chapter 2.1.2. These licenses are based on existing open source licenses but include additional paragraphs to regulate competing SaaS offerings, de-facto preventing commercial competition (Elasticsearch B.V., 2021; Igor Kotua, 2022; MongoDB, Inc., 2022; Timescale, Inc., 2020).
Single-vendor COSS companies can decide to release future versions of their software under changed licensing terms because they hold all of the copyright (Riehle, 2020).
Consequently, some of these companies stopped releasing their products under an open source license, and instead switched to cloud protection licenses, such as the RSAL or the SSPL. They explicitly named the competition by public cloud providers as their motivation. MongoDB Inc. for instance claimed that "it is too easy for large cloud vendors to capture all the value but contribute nothing back to the community" (MongoDB, Inc., 2022, first paragraph), when explaining their decision to switch to the SSPL.
Initially, the creators of cloud protection licenses positioned them as a modern approach in spirit of the open source principles, adapted to today's environment (Elasticsearch B.V., 2021; MongoDB, Inc., 2022; Timescale, Inc., 2020). However, the OSI did not accept any application of a cloud protection license (Open Source Initiative, 2021). One major reason was the violation of criterion 6 of the open source definition (Open Source Initiative, 2021) by cloud protection licenses. As cloud protection licenses restrict others from offering the licensed software as a service, they restrict the application of the software in this specific field. Regarding the SSPL, MongoDB withdrew the application at the OSI when they realized that the license would not be recognized as open source license (Open Source Initiative, 2021). Consequently, cloud protection licenses are not open source licenses because they lack the acceptance of the OSI (Elasticsearch B.V., 2021; MongoDB, Inc., 2022).

2.4 Open Source Community Health

Various aspects, perspectives, and terms are used to examine the state of open source projects. Goggins et al. (2021) define "open source project health" as the overarching concept. In this context, they see open source project health as "a project's ability to continue to produce quality software" (Goggins et al., 2021, p. 1). Linåker et al. (2022) name the top-level concept "open source software health" and define it as the "capability to stay viable and maintained over time without interruption or weakening" (Linåker et al., 2022, p. 1).
Goggins et al. (2021) identify three main perspectives on open source project health: success, sustainability, and survivability. Furthermore, researchers also use the terms risk and health itself to cover areas of these perspectives (Goggins et al., 2021). All these terms are used to investigate all parts of open source project health, although the perspective and term a researcher adopts indicates an initial focus on a part of the complex construct of an open source project. Researchers who investigate the success of a project usually focus on outcome related metrics, for instance (Goggins et al., 2021). For sustainability,

the development of accessible resources and coordination is more important Goggins et al. (2021). Survivability on the other hand indicates a focus on resilience and is more closely related to risk Goggins et al. (2021).
All in all, the terms used in open source research are vaguely defined.
The thesis follows Goggins et al. (2021) and defines "open source project health" as the top-level concept. For this purpose, "open source project health" is clearer than "open source software health" used by Linåker et al. (2022) because the wording of the first includes all aspects of the open source project while the latter indicates a focus on software aspects.
Figure 2.1 provides a non-exhaustive hierarchical view of the concept of open source project health derived from the work of Goggins et al. (2021) and Linåker et al. (2022).

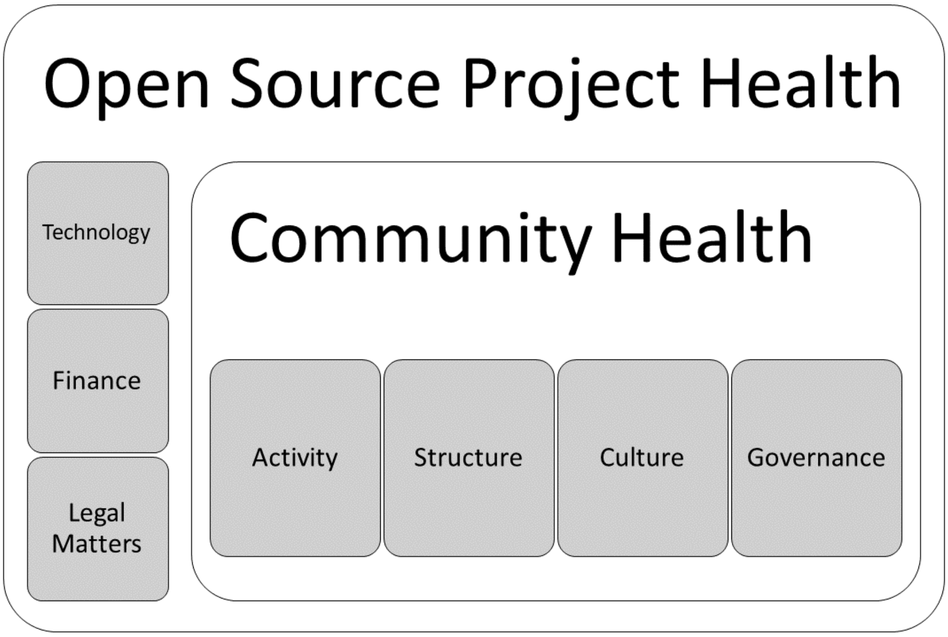

Figure 2.1: Hierarchical view of open source project health and open source community health

A first important perspective on open source project health focuses on purely technological aspects, such as the complexity and quality of the software, or security issues because of software dependencies or vulnerabilities.
The financial resources available to the project represent another element influencing its health. However, in case of COSS projects there is financial backing from the respective company. Consequently, this perspective is less important compared to traditional OSS projects were financial resources are often scarce.
Legal matters can be of decisive influence for the project. Researchers are in particular interested in the legal implications of different open source licenses which define how the software can be used.
This thesis focuses on "open source community health" which we define as the part of open source project health that focuses directly on the community. As displayed in

Figure 2.1, it includes the topics of activity, structure, culture and governance. Researchers examining the culture in open source communities investigate for instance the sentiment in messages between community members or how new members are welcomed.
Regarding the governance of communities, researchers examine whether governance structures are defined and whether the processes are followed in practice.
This thesis focuses on the aspects of community activity and community structure to evaluate community health. There are two main reasons for this decision. First, to keep the scope feasible, the research must focus on certain aspects. Second, community activity and community structure are the most examined aspects of community health but there is no research yet about their relation to cloud protection licenses.
Correspondingly, research on community activity and community structure is presented in more detail in chapter 3.1 and chapter 3.2.

2.5 Open Source Software Development

Some terms specific to software development, and open source software development in particular, will be used throughout the thesis. This section introduces those terms.
An open source project is the construct in which multiple parties collaborate to create software (Hauge & Ziemer, 2009; Riehle, 2020). In the context of this thesis, it is relevant to note that the open source project and the single-vendor COSS company which controls the project are two separate entities (Riehle, 2020).
In the project, multiple companies can collaborate. Similarly, a company can be active in multiple open source projects. Single-vendor COSS companies usually develop their product in an open source project with the identical or very similar name as the company. Nevertheless, they are separate entities.
Software developers use version control systems to have a fine grained control over the history of changes applied to the software (Blischak et al., 2016). When a developer wants to add her changes to the history, she commits them to the version control system. The term "commit" therefore describes the action of storing changes permanently. Usually, and in particular in open source software development, those changes are then made available to all other developers participating in the project (Blischak et al., 2016).
To manage the software development process, teams use project management systems (Dhasade et al., 2020). In such systems, entitled team members can create "issues" which represent a task that should be addressed (Dhasade et al., 2020). An issue can describe a new feature that should be added to the software, a defect of the software that should be fixed, or other tasks (Dhasade et al., 2020).

3 Theoretical Background

This chapter summarizes relevant findings from existing research in the field of open source software. First, it presents research about open source community health as defined in chapter 2.4. For this purpose, chapter 3.1 explores research about community activity. In chapter 3.2, literature about community structure is presented. Last, chapter 3.3 introduces existing research about the effects of license choice and license change in open source projects.

3.1 Activity in Open Source Projects

Researchers evaluate multiple types of activities in open source projects (Goggins et al., 2021; Linåker et al., 2022). Activity related characteristics are among the most commonly investigated characteristics of the projects, partly because they are easy to measure (Goggins et al., 2021). Each type of activity can be subdivided into more detailed aspects. As a consequence, researchers define diverse taxonomies of community activity (Cheng & Guo, 2019; Wang et al., 2020, e.g.). In favor of simplicity, this thesis distinguishes only two types of activity on the highest level, technical activity and social activity.

Technical activity is represented by actions which contribute directly to the project by resulting in changes of files of the software project. Typical technical activities are code contributions, i.e. changing the actual software, or non-code contributions, e.g. updating an associated image file.

Several researchers focus only on technical activities. Colazo and Fang (2009) aim to evaluate the relation between software licenses used and several properties of open source projects. One of these properties is coding activity, a type of technical activity. They created a formula to calculate an activity score which takes into account the lines of code added or deleted per commit, the number of commits, and the number of core developers. They extracted the required information from the version control system of the projects and evaluate the score for each quarter. As a key result regarding coding activity, they found that coding activity is higher in projects using copyleft licenses compared to those using permissive licenses.

As a building block of their research, Valiev et al. (2018) investigate technical activity to identify dormant projects using data retrieved from GitHub. In their approach, a dormant project is defined by a low level of technical activity in the year before the most recent activity event, i.e. "less than one commit per month on average in the 12 months prior to its most recent commit" (Valiev et al., 2018, p. 4).

Similarly, Gamalielsson and Lundell (2014) evaluate the development of the number of monthly commits for three related open source projects as part of their effort to characterize the projects. They retrieved the required data from the version control systems of the projects.

Social activity on the other hand describes communication between community members which contributes only indirectly to the project. Examples are the exchange of coordinating messages between community members or communication activities targeted at an audience outside of the open source project.

Often, the analysis of social activity is combined with examining some form of technical activity.

Wang et al. (2020) evaluate data retrieved from GitHub to investigate which activities are performed by "elite developers", those community members who hold administrative rights. GitHub provides records of a variety of events which can be associated with individual community members. By searching for those events which require administrative rights, Wang et al. (2020) identify the associated community members as elite developers. Then, they classified activity events from GitHub into four categories, communicative, organizational, typical and supportive activities, to evaluate which activities elite developers perform. As a first first key result, Wang et al. (2020) found that elite developers are responsible for the majority of activity in three of the four categories, with the exception of communicative activities where they still performed 34%. As a next step, they investigated the monthly development of the activities of elite developers and found that they increasingly perform communication and supporting activities. Furthermore, Wang et al. (2020) correlate the development of activities of elite developers with measurements of the outcome of the respective projects. Here, they found that the increase in communication and supporting activities is correlated with a loss in productivity in terms of commits and bug fixes. Regarding the quality of the software, Wang et al. (2020) state that the increase in communication and supporting activities correlates with more detected defects per month (which they see as negative effect on quality) and at the same time with a higher fix rate of those defects.

Cheng and Guo (2019), for instance, evaluated several types of community activity to determine the role an individual plays in the community. Building on four different types of contributions (code contribution, opinion contribution, network, and admin), they compose six types of activities. Then, they assign each community member one of 9 roles for each quarter, depending on the type and amount of activities the member performed in this time period.

Aué et al. (2016) investigated the relation between the success of an open source project and its diversity in terms of gender and geographical background of the community members using data from GitHub. They consider four metrics to determine the success the project, the growth of the team and three activity metrics. This way, they evaluate both, technical activity by analyzing the growth of commits and social activity by evaluating the growth of pull requests and comments.

Measuring community activity is also criticized by researchers. Shaikh and Levina (2019) claims that it can only be the starting point of an analysis but other aspects must be included to generate insights.

3.2 Structure of Communities in Open Source Projects

Researchers analyze many different aspects related to the structure of open source communities (Goggins et al., 2021; Linåker et al., 2022).

A common approach is to derive structural information from individual activity behavior by assigning roles to community members.

In the already introduced work of Cheng and Guo (2019), the authors developed a fine-grained role taxonomy which consists of four active and five supporting roles and assigned each community member a role depending on the type and amount of activities the member performed.

More commonly, the simpler "onion model" is used to evaluate the structure of open source communities (Amrit & Van Hillegersberg, 2010; Nakakoji et al., 2002). It describes the presence of few core contributors, individuals who contribute often, surrounded by layers of decreasingly involved community members.

Nakakoji et al. (2002) identified three different types of open source projects based on their aim and analyzed the roles of community members in projects of each type. The structure of the community in terms of distribution of the roles varies depending on the type of project. Communities in exploration-oriented projects try to innovate in the field of software development. In these projects, there is a small number of highly involved members who develop the software. Additionally, a few outsiders are occasionally involved to provide feedback. Utility-oriented projects aim to provide a very specific functionality primarily for use by the developers. Consequently, the community is small, but every member is highly involved. Peripherical members usually do not exist as their small and occasional contributions would be ignored by the core team that simply follows its own plans. Last, service-oriented projects target a large group of users. Therefore, they have fewer core developers and a large number of members at the periphery.

Amrit and Van Hillegersberg (2010) determine the role of community members depending on the files they worked on. First, they establish an order consisting of nine clusters of files describing different levels of how central these files are to the software project. According to the amount of work community members perform on files in the clusters, they assign the members a role between core and periphery for different points in time. They proceed by analyzing how these roles change and identified patterns in different projects. According to them, a healthy community will show movements towards the core, an unhealthy community will show tendencies towards the periphery. Furthermore, there are projects which show oscillatory movements between core and periphery, which they interpret as unstable situation.

Another group of researchers analyze how many individuals join or leave the community, for instance Aué et al. (2016) and Gamalielsson and Lundell (2014). Aué et al. (2016) quarterly evaluate the growth of the community as one of their metrics to determine the relative success of open source projects. They classify the projects into five groups corresponding to their quarterly team growth, with each group representing a certain percentile of projects. Gamalielsson and Lundell (2014) on the other hand investigate the retention of contributors as another property to characterize the evolution of three examined open source projects by tracking the monthly numbers of new, departed and total community members. Joining community members are identified by the first record of a commit in the version control system. Similarly, they recognize leaving community members by their last commit. The total number of members is the difference between all joined and all departed members.

Combining research on roles of community members and leaving members, some authors evaluate the so called "bus factor" (sometimes also "truck factor") (Cosentino et al., 2015; Torchiano et al., 2011). This factor represents the number of critical members that could lead to a failure of the project if they would leave the community, or get hit by a bus (Cosentino et al., 2015; Torchiano et al., 2011). Correspondingly, a low bus factor means that very few members leaving could already lead to a failure of the project, and therefore indicates a high risk (Cosentino et al., 2015; Torchiano et al., 2011). A larger bus factor indicates a greater resilience in this regard (Cosentino et al., 2015; Torchiano et al., 2011). There are different ways to determine the bus factor for a project. Cosentino et al. (2015) determine the bus factor using meta-data from Git to calculate each individual's share of contribution to certain files and the overall project. Torchiano et al. (2011) associate each individual with all the files he or she worked on. Additionally, they define a proportion of the total amount of files in the project which must be associated with at least one individual. Then, they calculate the bus factor by determining the minimum number of members that must leave so that less than this proportion of files is associated with remaining community members.

A separate topic of interest is the individual background of community members. For instance, whether community members are volunteers who contribute without being paid, or employees who take part in the project because of their employers (Claes et al., 2018; Riehle et al., 2014). Riehle et al. (2014) identify contributions made by paid members using the timestamp of the contribution. Every contribution provided in working hours counts as paid contribution. This way, they calculate that around 50% of all work in the analyzed project was performed by employees.

Considering COSS projects in particular, one aspect investigated is whether members belong to the company which owns the project or whether they are part of an external organization (Dias et al., 2018). Dias et al. (2018) further analyze the proportion of work performed by members internal or external to the company and the roles they fulfilled. Their key findings are that the proportions vary across projects and internal and external members both perform all types of activities. Nevertheless, internal members are usually more involved in coordinating tasks and remain important regardless of the share of work performed by external members. Most external members provide only few contributions but in some cases they also fulfill coordinating roles.

3.3 License Choice and License Change

Selecting a license for an open source project is a crucial decision with far reaching implications. The license sets the legal boundaries regarding the usage of the software (Wilson, 2013). For many individuals, the choice of license impacts the decision to become an active member of the community of a project (Colazo & Fang, 2009; Stewart et al., 2006). According to Colazo and Fang (2009), copyleft licenses motivate volunteers to contribute to open source software. The volunteers see themselves as part of a social movement and copyleft licenses preserve the ideals of this movement. Stewart et al. (2006) reason that developers are motivated to contribute to OSS projects because they use the resulting software themselves (utility motives), or because they can increase their reputation and visibility to enhance their career, or out of idealism. Depending on the presence and type of sponsorship of a project, the type of license might impact these goals (Stewart et al., 2006).

Researchers investigate the relation between the type of license and various characteristics of open source projects. Most researchers distinguish only between copyleft and permissive licenses as described in chapter 2.1.

In their work, Colazo and Fang (2009) found support for larger communities as well as higher technical activity in projects using copyleft licenses. Furthermore, those projects with copyleft licenses released more frequently compared to those using permissive licenses. Last, they could find a significant relation between type of license and the duration of memberships in the community but contrary to their expectations, developers remain longer members in projects with permissive licenses.

Stewart et al. (2006) examined the impact of sponsorship and the restrictiveness of the license on technical activity and user interest. To analyze the impact, they selected several OSS projects which show different characteristics. First, whether there is an organization sponsoring the project, and if so whether this organization has commercial interests. Here, educational and government organizations were named as such without commercial interests. Second, whether a copyleft or permissive license is used by the project. When there is no sponsor at all, there is no big difference between permissive and copyleft licenses, but the latter might protect the ideals of open source. When the sponsor has commercial interests, a copyleft license ensures the open source nature of the project in the future. Moreover, according to Stewart et al. (2006), copyleft licenses also protect utility and reputation motives. When a commercial sponsor uses a permissive license for the project it could become a standard which is controlled by the sponsor. Single developers fear that they then cannot modify the software according to their needs, which decreases the utility-based motivation. Furthermore, the project is then strongly associated with the sponsor which decreases the visibility of the individual contributor. Therefore, Stewart et al. (2006) argue that developers prefer copyleft licenses in particular in those cases when there is a commercial sponsor.

When the sponsor of the project has no commercial interests, developers do not fear that

the project will show these problems regardless of which license is used. In this case, they prefer a permissive license which increases the flexibility of the project.
When testing these hypotheses, Stewart et al. (2006) found that in overall the influence of the choice of copyleft or permissive license on technical activity was not significant. However, it was significant when only projects with no sponsor or a commercially oriented sponsor were considered. In case of projects with a non-commercially oriented sponsor the relationship was even the opposite than expected, which offsets the overall view.
Stewart et al. (2006) suggest that this is the case because a non-commercially oriented sponsor already ensures the adherence to open source ideals. Then, developers prefer the flexibility of permissive licenses (Stewart et al., 2006).
In spite of this, Vendome et al. (2017) identified a trend towards permissive licenses when examining open source projects on GitHub. In line with this trend, when projects change licenses, they tend to switch to less restrictive licensing terms.
In the four cases Viseur and Robles (2015) studied, the motivation for the change ranged from improving collaboration with other parties to plain simplification of existing terms. The new licensing terms can be advantageous for the project and lead to wider distribution, increased revenues, or the abandonment of competing forks. On the other hand, they can cause incompatibility with other projects.
Santos (2017) investigated the relationship between changes of software licenses and attractiveness which he defined as a combination of user demand and the number of community members. In a first step, he confirmed that license changes indeed occur. Then, he categorized combinations of licenses into license schemes according to the set of rights they provide, deviating from the simple distinction between copyleft and permissive licenses. Next, he analyzed how frequent certain types of changes are, e.g. from no license at all to a copyleft license. Finally, he evaluated the impact of each type of change on the attractiveness of OSS projects. As a result he could show that changing the license affects the attractiveness of projects. Whether it is a positive or negative impact, and how big the impact is, depends on the type of change.
Furthermore, the change can be undesired by the existing community and lead to decreased motivation. If there are enough unhappy community members, they could fork the project which weakens the original community and creates a new competitor.

4 Propositions

The fourth chapter builds on the theoretical and domain background presented in chapter 2 and chapter 3. It develops expectations for the impact of the license change on the community, answering the first and second subquestion. These expectations will be compared to the results of the analysis of the projects in chapter 7.4.

As we could see in chapter 3.3, various researchers investigated the impact of the choice of license on open source projects. Closely related to the choice of license is a change of licensing terms. The difference between the two topics in research is that in the first case different projects using different licenses are compared at the same point in time (Colazo & Fang, 2009; Stewart et al., 2006), while in the case of license changes the state of a project is compared to the state of the same project at different points in time (Santos, 2017; Vendome et al., 2017; Viseur & Robles, 2015). However, the underlying mechanisms which determine the impact of license choice and change are identical.

Similar to the work of Colazo and Fang (2009) and Stewart et al. (2006), we take the individual community member as a starting point to develop propositions. According to them, individual persons get involved into open source projects for various reasons. Some enjoy being part of a social movement and are driven by ideological motives (Colazo & Fang, 2009; Gamalielsson & Lundell, 2014). Others contribute for the pleasure of the work itself, or want to improve the software they use (Colazo & Fang, 2009; Gamalielsson & Lundell, 2014; Stewart et al., 2006). Some members join communities to build a reputation and enhance their careers (Stewart et al., 2006). Additionally, for some contributors it is simply part of their job to work in an open source community (Riehle et al., 2014).

Furthermore, in chapter 3.3 we described that changes in licensing terms affect the attractiveness of an open source project (Santos, 2017) and can be undesired by the community (Viseur & Robles, 2015).

Changing the license from an open source license to a cloud protection license is a clear step away from the ideals of open source software. Therefore, we can expect that this change will affect the motivation of individuals to be part of the community of the respective project. By impacting persons from the community individually, the change of license impacts the community as a whole.

Figure 4.1 displays the causal diagram of the underlying conceptual model. The independent variable is the license change. We expect the license change to have an impact on the dependent variable, community health.

As explained in chapter 2.4, we will focus on community activity and the structure of the community as proxies for community health.

Next, the expectations regarding the activity of the respective communities are formulated, followed by those regarding the structure of the communities.

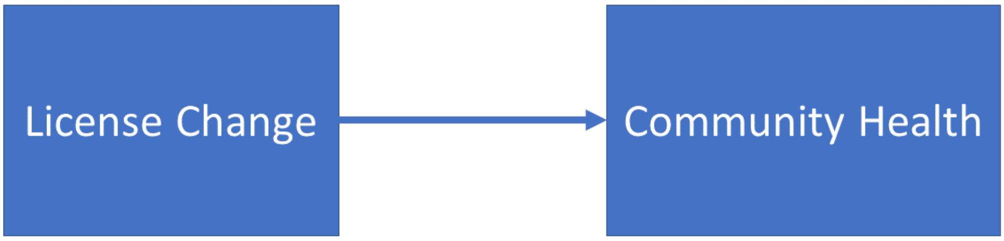

Figure 4.1: Conceptual model

4.1 Community Activity

The research of Colazo and Fang (2009) and Stewart et al. (2006) is based on the idea that open source community members are driven by the ideals of open source software. Similarly, Gamalielsson and Lundell (2014) recognized ideological motivations among open source community members. According to the findings of Colazo and Fang (2009), there is more activity in open source projects using copyleft licenses than in those with permissive licenses. As described in chapter 2.1, copyleft licenses are more aligned with ideological motivations because they ensure that derived software remains open source. As stated before, switching from an open source license to a cloud protection license is a step away from the ideals of open source software. Combining these insights, we therefore expect reduced community activity when a company changes the license of its software from an open source license to a cloud protection license.

> *Proposition 1: Changing the license from an open source license to a cloud protection license leads to reduced community activity.*

4.2 Community Structure

It is normal in an open source community that there are new members joining the community and some members leaving the community for various reasons (Cheng & Guo, 2019). Several researchers studied the relation between the software license in use and characteristics of this process, as described in chapter 3.3. According to the work of Colazo and Fang (2009), the time period between joining and leaving a community correlates with the license. Santos (2017) showed that a change of licensing terms correlates with the

attractiveness of open source projects, which in turn influences the decision to become a member of the community. However, his study of changes between specific "licensing schemas" does not allow to draw conclusions for the specific license change studied in this thesis.
Consequently, we again base our expectations on ideological motivations of community members (Colazo & Fang, 2009; Gamalielsson & Lundell, 2014; Stewart et al., 2006). Similar to chapter 4.1, we expect that there will be less persons joining a community after it switches to a cloud protection license because it will reflect the ideals of open source software to a lesser extent.

> *Proposition 2: Changing the license from an open source license to a cloud protection license leads to less persons joining the community.*

Similarly, we can expect more community members to make the decision to leave the community.

> *Proposition 3: Changing the license from an open source license to a cloud protection license leads to more persons leaving the community.*

Cosentino et al. (2015) and Torchiano et al. (2011) pointed out that in open source projects there is a concern of dangerous knowledge concentration when key members leave the project. We argue that those persons who leave the community because of a license change to a cloud protection license are very involved members because these are motivated by open source ideals.. As a result, the concentration of knowledge among the remaining community members would increase.

> *Proposition 4: Changing the license from an open source license to a cloud protection license leads to increased knowledge concentration considering individuals in the community.*

Given that there was a public dispute with executives from collaborating companies announcing their disagreement with the new licensing terms (Igor Kotua, 2022), we can also expect that there are companies which stopped contributing to the projects. Therefore, we can expect the concentration of knowledge regarding the involved organizations to increase.

> *Proposition 5: Changing the license from an open source license to a cloud protection license leads to increased knowledge concentration considering organizations in the community.*

5 Methodology

In this chapter, the methodology to examine the propositions formulated in chapter 4 is presented. In the beginning, it justifies the selection of Community Health Analytics in Open Source Software (CHAOSS) as primary tool. Subsequently, it provides a summary of the data sources used in the thesis and the methodologies to examine the propositions. Figure 5.1 displays an overview of the interactions between data sources and tools. We use CHAOSS to collect data from the two data sources Git and GitHub. CHAOSS then enriches the collected data. For this thesis the enrichment is relevant to associate contributions with individual community members. Moreover, CHAOSS offers simple visualizations out of the box which we use to create Python scripts to produce more sophisticated visualizations.

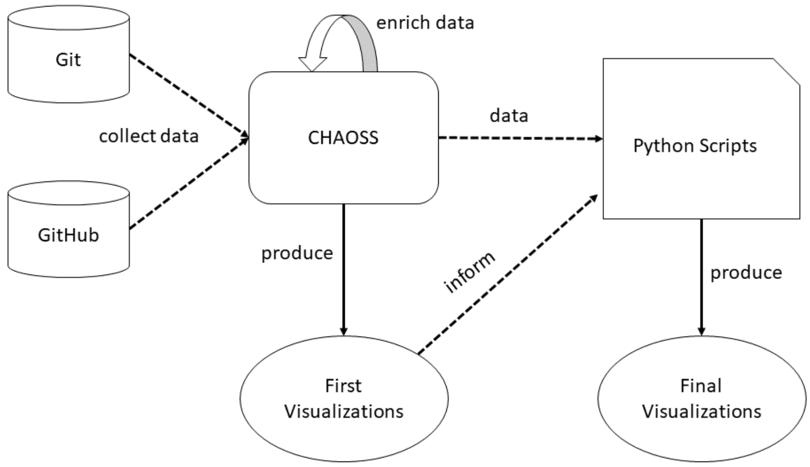

Figure 5.1: Diagram summarizing the steps of the analysis

5.1 Selection of Tools

Researchers investigating open source software often pair their research with the development of tools to perform it. Amrit and Van Hillegersberg (2010) for instance developed TESNA, a tool to analyze socio-technical patterns in open source communities. Cosentino et al. (2015) also developed their own tool to perform their specific approach. Those tools are usually also open source and therefore available for other researchers.
Evaluating OSS projects is not only an academic problem but also a concern of practitioners. Consequently, they also created methodologies and tools to facilitate this process. Arne-Kristian Groven et al. (2011) provide a comprehensive overview of different approaches, such as QualOSS, QSOS, OSMM, or OpenBRR. Each methodology defines a

set of characteristics and corresponding metrics which should be analyzed to assess the state of the OSS project.
In order to examine the propositions formulated in chapter 4, tools are required to extract and analyze the corresponding data. Developing a custom tool to perform these tasks is out of scope of this thesis.
Recently, the Linux Foundation created the CHAOSS project to work on the evaluation of OSS projects (Linux Foundation, 2022b). CHAOSS bundles the definition of metrics with the development of software to measure them. In contrast to previous approaches, CHAOSS does not define a methodology about which of its metrics should be taken into account. Furthermore, CHAOSS is the most recent project and provides tools which were developed to work with the modern development environment. CHAOSS consists of a set of tools and offers a simple way to use them. GrimoireLab (Dueñas et al., 2021) is the entry point for users. Additional tools facilitate the retrieval of data or mapping of identities. After providing basic configuration, the tools automatically provide visualizations of data which allow to draw first conclusions and refine the methodology accordingly. For finer grained analysis, CHAOSS offers interfaces which can be used to extract raw and transformed data. Furthermore, CHAOSS allows to collect data from many different data sources, essentially most platforms commonly used in open source development can be set up as data source. This offers a lot of flexibility and completeness while the effort necessary to learn how to use the tools is comparable to other tools.
The tools of CHAOSS are used to evaluate projects in this thesis. In the following, we will refer to them using the term CHAOSS even when strictly speaking it is one of the components of CHAOSS which is used. They were configured to retrieve the corresponding data and their visualizations were used to analyze the data but also refine the methodology. Additionally, data was extracted and analyzed using Python scripts.

5.2 Data Sources

The data analyzed in this thesis is extracted from two types of data sources.
First, the Git repositories of the projects. As described in chapter 2.5, software developers use version control systems to remain in control over the history of the software project, i.e. be able to undo and reapply changes at any time. Git is a popular version control system (Blischak et al., 2016). To enable version control, Git stores the complete history of commits in the project. Each commit is associated with meta-data such as the author's name and email address, the time of contribution, and a descriptive message (Blischak et al., 2016). Consequently, a Git repository is the single source of truth regarding the technical development of a software project because it contains a timeline of all changes ever committed to the source code. While theoretically the timeline can be changed, this is almost never done in practice to ensure the integrity of the history. The data stored in a Git repository therefore represents factual information about the evolution of the corresponding software project.

Git can be used to analyze various aspects of software projects, in particular because of the associated meta-data. It can be used to analyze the activities of individual community members, but also to draw conclusions for the community as a whole.
As an example, Riehle et al. (2014) used the associated timestamp of the commits stored in Git to identify whether the commit was performed during working hours. Building on the results, they evaluated the share of commits performed during working hours for several software projects and therefore examined a characteristic of the community in its entirety. Furthermore, Riehle et al. (2014) combined the timestamps of the commits with the information about the author to categorize individual members either as paid employee who works on the software during working hours, or as volunteer who works outside of working hours.
In conclusion, Git is an essential tool for software development and contains valuable factual information about the development of the respective software project. Consequently, Git (or similar version control systems) is a widely used data source for researchers investigating open source software (Claes et al., 2018; Colazo & Fang, 2009; Cosentino et al., 2015; Gamalielsson & Lundell, 2014; Riehle et al., 2014; Torchiano et al., 2011).
In this thesis, we use the timeline of commits stored in Git to calculate technical activity, as described in chapter 5.3.1, and to determine the number of monthly joining and leaving community members, as described in chapter 5.3.2. Furthermore, CHAOSS relies on Git to associate community members with an onion role and an organizational background, both will be discussed in chapter 5.3.2. The Git repositories of the projects analyzed in this thesis are hosted on GitHub, in order to make them publicly available.

In addition to hosting the Git repositories, there are further information available on GitHub which are independent from Git (Dhasade et al., 2020). We use these additional information as second type of data source.
GitHub is a software development platform which contains one of the largest collections of software projects in the world (Dhasade et al., 2020). These software projects, host their Git repositories on GitHub, just as the projects examined in this thesis (Dhasade et al., 2020). As a development platform, not only the corresponding Git repository of the software is available, but also a lot of information around the development process itself, such as accompanying communication and project management (Dhasade et al., 2020). A typical workflow on GitHub is that an issue is created related to a defect of the software and community members exchange messages to plan how to work on that issue (Dhasade et al., 2020). Subsequently, community members works on the source code and propose a change which addresses the issue (Dhasade et al., 2020). Then, the community discusses whether the changes are sufficient and in a good state (Dhasade et al., 2020). There might be multiple iterations of discussion and actual work, before eventually the changes are accepted (Dhasade et al., 2020).
Records of these activities are stored on GitHub and can be retrieved. Correspondingly, this data represents factual information about the collaboration in software projects. Re-

searchers from various disciplines make use of data from GitHub (or similar platforms) because of the rich information it provides (Aué et al., 2016; Cheng & Guo, 2019; Claes et al., 2018; Dias et al., 2018; Valiev et al., 2018; Wang et al., 2020). Some even rely solely on GitHub as a data source for their research, for instance Cheng and Guo (2019) and Wang et al. (2020) as described in chapter 3.1.

In this thesis, we use the record of created issues stored by GitHub to calculate social activity within the software projects, as described in chapter 5.3.2. All three companies use GitHub for development, therefore we use this publicly available data to analyze their projects. The references to the respective projects are provided in chapter 6.

5.3 Time Series Analysis

The basic approach for investigating the impact of the license change is the evaluation of time series data. For each project, public data from the beginning of the respective project until the end of 2022 is collected. This raw data is transformed into the metrics described in chapter 5.3.1 and 5.3.2. Then, the transformed data is visualized. As we know the dates on which the respective projects changed their licensing terms, we can inspect the visualizations to see whether there are changes around these dates.

However, to identify possible changes it is important to take the nature of the data into account. If it shows seasonal fluctuation this should be considered. If the absolute value of the metric varies strongly across the time range this should be considered because we are interested in changes in the metric, not the absolute value.

A common way to integrate these considerations is to decompose time series data. Cleveland et al. (1990) introduced the seasonal-trend decomposition based on loess (STL) methodology which we adopt. Equation 5.1 illustrates the underlying idea, where y represents the observed data, s represents a seasonal component, t represents a trend component, and r is the residual term.

$$y = s + t + r \tag{5.1}$$

After decomposing time series data, long-term developments can be observed in the trend component. The residual component on the other hand indicates exceptional events, i.e. the component of the data which cannot be explained by seasonal effects or the long-term trend.

5.3.1 Community Activity

As described in chapter 3.1, there are different types of activity in an open source community that are evaluated by researchers. The two main categories are technical activity and social activity which both consists of finer grained subtypes. In this thesis, there is no reason to limit the analysis on a specific type of activity. Instead, for both main types of

activity data is available in the data sources and therefore both types are included in the analysis.
In CHAOSS, technical activity is represented by the metric "code changes commits" which counts commits as described in chapter 2.5. This data is available in Git. Each commit is associated with an exact point in time when it was performed. We aggregate commits per month to observe the development over the history of the projects.
In the metrics defined by CHAOSS, "collaboration platform activity" corresponds to social activity (Linux Foundation, 2022a). Regarding the selected projects, GitHub is the collaboration platform. We select the number of created issues on GitHub per month as metric for social activity.
Both metrics, monthly commits and monthly created issues, are extracted by CHAOSS from Git and GitHub respectively. Using a Python script, we then read the metrics from CHAOSS and decompose both time series data using the STL method and visualize the results. As we use monthly data for both metrics, we expect a seasonal component which varies over the course of a year (periodicity of 12).
In chapter 4.1, we stated that we expect reduced community activity after the license change occurred. By inspecting the visualizations of the decomposed time series, we can examine whether the data is congruent with this proposition.

5.3.2 Community Structure

To evaluate the development of the structure of the community, we decided to investigate two aspects. First, we analyze the development of joining and leaving members. Second, we examine the concentration of knowledge within the community, considering individuals and organizations.
In the metrics defined by CHAOSS, "contributors", "new contributors", and "inactive contributors" provide insights about joining and leaving community members.
CHAOSS extracts the complete history of commits from Git and associates each commit with an individual member. In a Python script, we read this information from CHAOSS and find the first commit of each community member. Subsequently, we calculate the number of first commits for each month representing the number of joining members.
Similarly, we can identify leaving members by evaluating the commits. For each community member, we find the most recent commit using a Python script. If this most recent commit occurred at least 6 months ago, we assume the member has left the community in the month of the commit. In the following, we aggregate the number of leaving members per months.
We decompose both time series data using the STL method and visualize the results. Again, we assume a seasonal component with a periodicity of 12 because of the monthly data.
Using the visualizations, we can examine the expected outcome of less joining and more leaving members.

To evaluate whether the license change impacts the concentration of knowledge within the community, we apply the commonly used onion model (Amrit & Van Hillegersberg, 2010; Nakakoji et al., 2002).
For this purpose, we evaluate the individual role of each community member according to the onion model. In the version of the onion model defined by CHAOSS, each community member is either a core contributor, a regular contributor, or a casual contributor (Linux Foundation, 2022a). Core contributors are the originators of 80% of all contributions in a quarter, regular contributors provide 15% of the contributions, and casual contributors the remaining 5% (Linux Foundation, 2022a), as displayed in Table 2.

Onion Role	Share of Contributions
Core Contributors	80%
Regular Contributors	15%
Casual Contributors	5%

Table 2: Shares of contributions per onion role in the model defined by CHAOSS

For each community member, CHAOSS determines the onion role for each quarter. Using a Python script, we read this information from CHAOSS and calculate the proportions of onion roles among community members. Subsequently, we visualize the proportions per quarter. In this case, we do not decompose the time series. As we use quarterly data, we do not expect strong seasonal effects.
This way, we use the onion model as indicator for knowledge concentration. The share of core members represents the smallest proportion of community members who were responsible for 80% of the contributions in a quarter. If this share decreases, the knowledge becomes increasingly concentrated.
According to proposition 4 we expect an increased knowledge concentration after the license change. We examine this proposition by inspecting whether the visualizations show supporting data.
Besides the individual role, we also examine the knowledge concentration with respect to the organizational background of the community members.
Again, we rely on CHAOSS to collect the required information. CHAOSS is able to associate individual members with organizations by examining the domains of email addresses. Correspondingly, before performing the analysis we configured CHAOSS to associate the respective companies with their domains as described in chapter 6. CHAOSS then automatically enriches each commit with the authoring member and the organizational background of the member.
Subsequently, we read the enriched data from CHAOSS using a Python script and calculated the share of contributions associated with the COSS company owning the project for each month. Similar to the previous calculation, we do not decompose the times series because we do not expect strong seasonal effects.
The share of contributions authored by employees of the respective company serves as indicator of the concentration of knowledge within the community in respect to the involved

organizations. In chapter 4.2, we proposed the expectation of an increasing knowledge concentration. Using the visualization of the shares of contributions associated with the company, we can examine proposition 5.

6 Projects

This chapter introduces the three companies which will be analyzed to examine the expectations formulated in chapter 4. All of them changed the license of their software projects. In all projects they abandoned an open source license and adopted a cloud protection license instead. This was only possible because they follow the single-vendor model and therefore hold all of the copyright, as described in chapter 2.2. However, in each project the exact licenses differ. This chapter first provides a brief explanation of the project selection process. Then, the main products of the companies are introduced. For each company, the circumstances related to the license change are explained, including date, motivation, and which license was used before and which license is used after the change. Last, links to data sources for the analysis are listed. Table 3 summarizes the company descriptions.

Company	Date of License Change	License before Change	License after Change	Projects
MongoDB	16-10-2018	AGPLv3	SSPL	https://github.com/mongodb/mongo
Elasticsearch	10-02-2021	Apache 2.0	SSPL, ELv2	https://github.com/elastic/elasticsearch
Redis	15-07-2018 21-02-2019	AGPLv3	Apache 2.0 with Commons Clause RSAL	https://github.com/RediSearch/RediSearch, https://github.com/RedisGraph/RedisGraph, https://github.com/RedisJSON/RedisJSON, https://github.com/RedisLabsModules/redisml, https://github.com/RedisBloom/RedisBloom

Table 3: Summary of the description of the three companies and their software projects

6.1 Project Selection

One problem considering finding suitable projects to analyze is that cloud protection licenses are a very new phenomenon and not widely adopted yet. Furthermore, the focus of this thesis is on the impact of changing the license. Consequently, it is not sufficient to find projects using a cloud protection license but they must have changed their licensing terms from an open source license to a cloud protection license. Additionally, there is no complete list of all existing cloud protection licenses. In the end, the following approach was used to identify projects. First, a list of cloud protection licenses was created based on licenses identified in blog posts (Igor Kotua, 2022). Then, projects using this license were searched on GitHub using its advanced search feature (Github Inc., 2022). This way, a list of projects was composed, as shown in Table A1 in appendix A.

From this list, only those projects which changed their licensing terms from an open source to a cloud protection license were considered. Last, three projects were selected which have a relatively large community to allow for an analysis using statistics.

6.2 MongoDB

The main offering of MongoDB Inc. is its document-oriented database MongoDB. Document-oriented databases are very flexible because they do not require the user to define the format of the data beforehand but allow the format to evolve with the application. This database is offered as a service as "MongoDB Atlas". When using a database as a service, users do not need to configure the database, for instance to perform backups. Furthermore, they get benefits such as availability and scalability because MongoDB as the service provider takes care of it. The flexibility together with the benefits of SaaS made MongoDB a very successful product.

Starting in 2009, MongoDB was developed under the open source AGPLv3 license. In 2018, MongoDB Inc. created the SSPL to protect its SaaS business. Since 16-10-2018 the core project of MongoDB, i.e. the database, uses the SSPL. By 31-01-2023 around 1200 individuals contributed to MongoDB, performing around 106000 commits. This core project alongside with all the data evaluated for this thesis can be found on GitHub (https://github.com/mongodb/mongo). Email addresses of employees of MongoDB use the domains "mongodb.com" and "10gen.com", because the company was previously called 10gen. As described in chapter 5.3.2, we use this information to identify employees among the community members.

6.3 Elasticsearch

Elasticsearch is a widely used search engine, capable of searching through all kinds of data. Usually, Elasticsearch is combined with other tools, especially Logstash and Kibana, to provide a platform to search, visualize and analyze data. They have a SaaS offering, the "Elasticsearch Service". In 2015, Amazon started its "Amazon Elasticsearch Service" launching a debate over unfair practices and trademark infringement (AWS, 2015).

The development until 2021 took place under the Apache 2.0 license. Since version 7.11, it adopted MongoDB's SSPL in a dual licensing scheme with its own Elastic License 2.0 (ELv2). The license change affects the core product, Elasticsearch, and occurred on 10-02-2021. Until 31-01-2023 around 2400 community members contributed 130000 commits to the project. The Elasticsearch project is located on GitHub (https://github.com/elastic/elasticsearch). Employees of Elasticsearch have email addresses using the domain "elastic.co".

6.4 Redis

Redis is an in-memory data store. It was mainly developed by Redis Ltd. Regarding this study, Redis differs from the previous two companies for two reasons. First, Redis did not change the license of the core product "Redis". Instead, they changed the license of accompanying products which are called "RedisModules". These modules simplify using the core product by providing, e.g., search capabilities. The modules were developed under the AGPLv3 license.

Second, Redis changed the license of these modules twice. The first change took place on 15-07-2018. Then, the Apache 2.0 license with commons clause was applied to all the software projects. However, the new licensing terms caused confusion about the legal situation. Consequently, Redis changed the licenses again on 21-02-2019. From this day, the projects are licensed under the RSAL.

By 31-01-2023 about 230 contributors provided 11400 commits to all five projects considered. All software projects are hosted on GitHub, as shown in Table 3. Email addresses of employees of Redis use the domains "redis.com" and "redislabs.com".

7 Results

This chapter presents the results of the analysis of the projects.
For each project, the STL decomposition of the number of monthly commits and monthly created issues are discussed to evaluate technical and social activity respectively. Subsequently, the STL decomposition of monthly joining and leaving community members are analyzed. Together with the following evaluations of the development of the distribution of the onion roles among the community members and the share of contributions authored by employees of the respective company, this provides insights about the structure of the communities. At the end of the chapter, we discuss similarities and differences between the results of the projects. Furthermore, the results are compared with the expectations proposed in chapter 4. Correspondingly, we answer the subquestions 3 and 4 in this chapter.

7.1 MongoDB

Here, the results of the analysis regarding MongoDB are presented. As described in chapter 6.2, MongoDB changed its license on 16-10-2018. The results regarding community activity are discussed first, followed by those concerning the structure of the MongoDB community. Each section is concluded with a summary of the results.

7.1.1 Community Activity

Figure 7.1 and Figure 7.2 show the development of activity inside the community of the MongoDB project. Figure 7.1 represents the technical activity, measured as commits per month. The figure consists of four plots. The first plot shows the total count of commits per month. Thereafter, three plots show trend, seasonal and residual component according to STL decomposition respectively. Vertical lines indicate the date of license change, which is 16-10-2018 in the project of MongoDB. Table B1 in appendix B.1.1 contains the corresponding data.
Figure 7.1 and Figure 7.2 both display all available data from the creation of the project until 31-12-2022. However, the first commits in the MongoDB project are from October 2007, as shown in Figure 7.1. The first issues on the other hand were created in September 2010.
We can see that in a typical month there are between 300 and 1000 commits made to the MongoDB project. Especially in the beginning of the project until December 2008, there are much smaller numbers of monthly commits, ranging from only 6 commits in May 2008 to 72 in November 2008. A few months show more than 1000 commits, notably October 2014 when the number of monthly commits reached its maximum with 1220 commits.
The trend plot of Figure 7.1 demonstrates that the average number of monthly commits increased until it reached a maximum in 2015. Subsequently, the amount of monthly

commits essentially halved for the years from 2016 until 2020. In 2020 the monthly commits started to increase again.
In the seasonal plot, we can see a seasonal component which changes its shape over the years. In particular the years from 2019 to 2022 show a similar seasonal behavior with a characteristic plateau on a high level in the first half of the year. Moreover, the years from 2010 up to 2014 show a similar seasonal component, although in this case it consists of small alternations per month.
The fourth plot of Figure 7.1 shows residual commits which represent deviation from trend and seasonal component. We can see that the residual term mostly alternates around 0. There are a few exceptions, in particular in 2009 and 2010 when the maximum number of monthly residual commits occurred. The years since 2020 show a notably higher alternation in residual commits than the years before.
MongoDB changed its license on 16-10-2018, visualized by the vertical red line in the plots. In the first plot of Figure 7.1, we can see that the number of monthly commits was almost constant in the months following the license change. The trend, as calculated by the STL decomposition, was slowly increasing in that period of time. The residual commits do not show any significant behavior in the first year after the license change. Here, we can only observe a change starting in 2020 with an increased alternation as described above.

Figure 7.2 shows the social activity which is operationalized as the number of issues created per month. The figure consists of four plots. The first plot shows the total count of issues created each month. Thereafter, three plots show trend, seasonal and residual component according to STL decomposition respectively. Vertical lines indicate the date of license change, which is 16-10-2018 in the project of MongoDB. Table B1 in appendix B.1.1 contains the corresponding data.
The number of issues created each month is much smaller compared to the number of monthly commits. Figure 7.2 displays the maximum number of monthly created issues in October 2014, which corresponds to the maximum number of monthly commits in the MongoDB project. The STL plots in Figure 7.2 can be divided into two periods which show different behavior.
In the first period from the beginning and including 2015, usually between 15 and 35 issues were created per month.
The trend plot shows an increasing trend until the maximum is reached in October 2014, then it decreases throughout 2015.
The first period shows a seasonal pattern with a small number of created issues in the early months of each year, as displayed in the third plot of Figure 7.2. Furthermore, there is a peak in the second half of every year until 2016.
Regarding the residual plot of the new issues in the MongoDB project, we see that in the period until 2016 there is a relatively high alternation.
In the second period starting in 2016, the number of monthly created issues is never

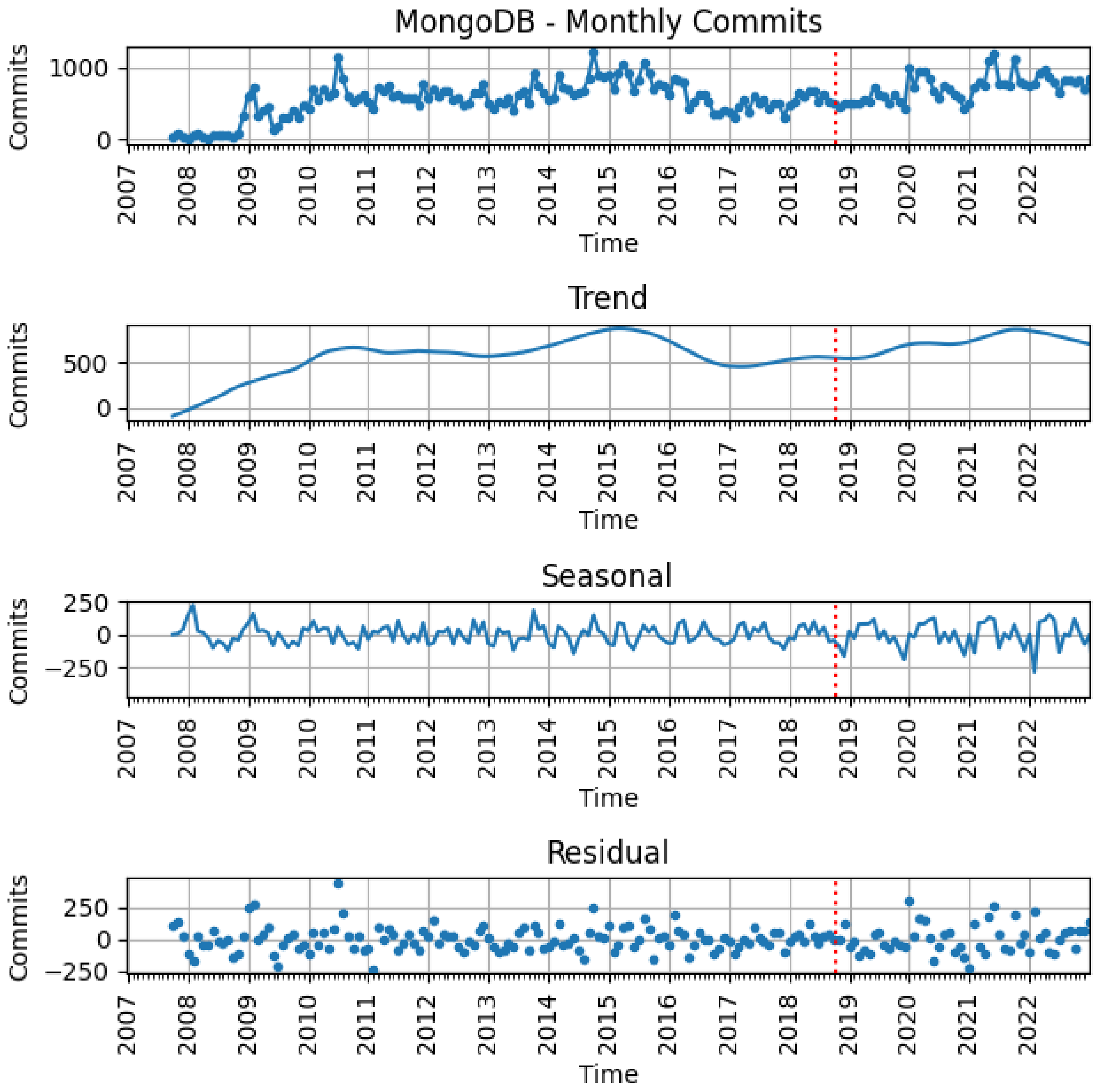

Figure 7.1: STL decomposition of monthly commits in the MongoDB project

greater than 12, and therefore much smaller than in the previous period.

Consequently, the trend remains at a low level, as displayed in the second plot of Figure 7.2.

The seasonal pattern remains similar in shape but as a consequence of the smaller numbers of created issues has a smaller amplitude.

In the second period, the residual monthly created issues show less alternation and approximate 0.

The license change of MongoDB falls into the second period. Therefore, the number of monthly created issues was small around this date and remained small afterwards. Figure 7.2 shows no indications of a change introduced by the license change in trend, seasonal, or residual plot.

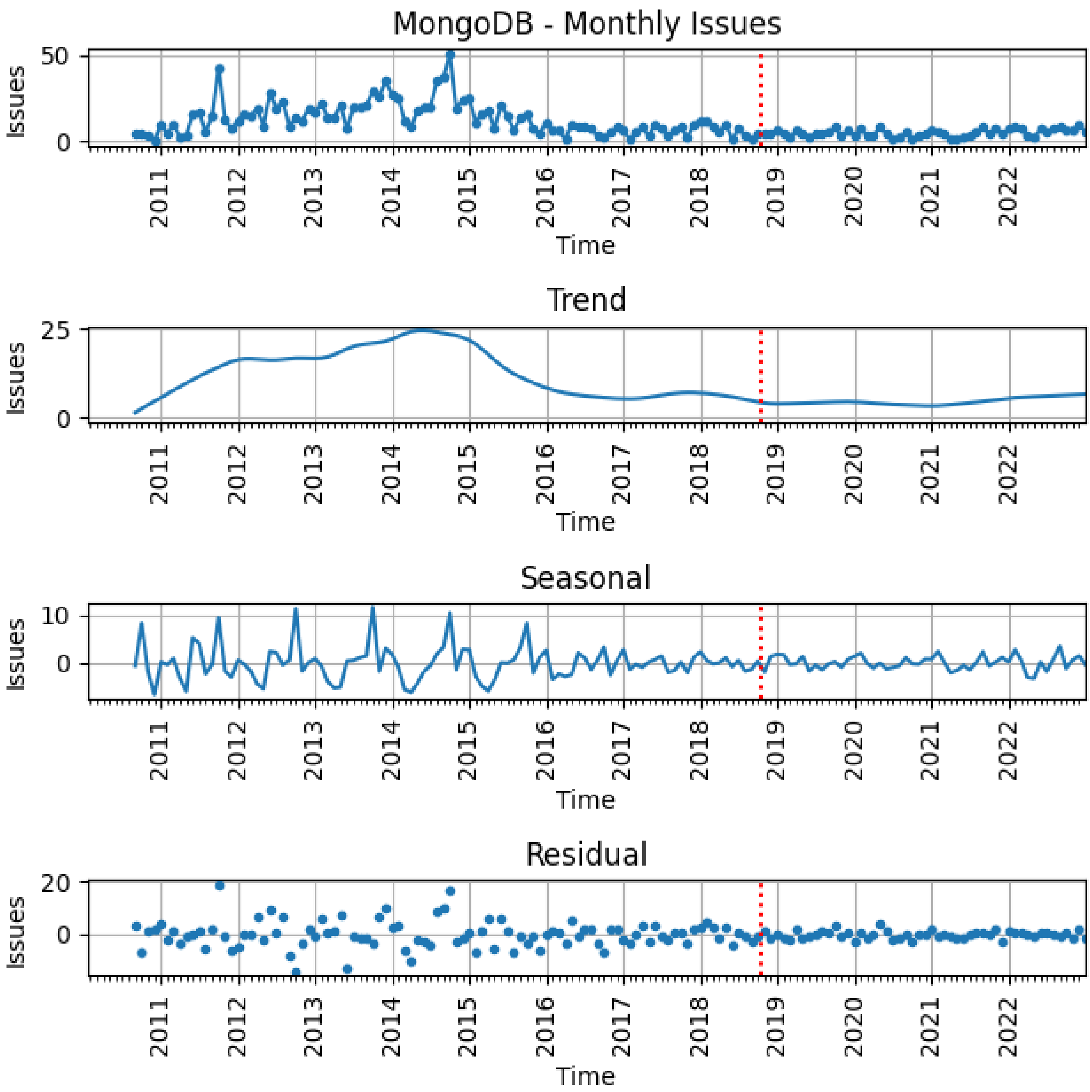

Figure 7.2: STL decomposition of monthly created issues in the MongoDB project

7.1.2 Community Structure

Figure 7.3 shows the development of individuals joining the community of MongoDB from the start of the project until 31-12-2022. The figure consists of four plots. The first plot shows the total count of monthly joining members. Thereafter, three plots show trend, seasonal and residual component according to STL decomposition. Vertical lines indicate the date of license change, which is 16-10-2018 in the project of MongoDB. Table B2 in appendix B.1.2 contains the data.

The number of monthly joining members in the MongoDB community is small in the beginning of the project. We can observe an slowly increasing trend from 2008 to 2012. In the following the trend component remains almost constant, alternating between values of 5 and slightly less. Starting in the second half of 2019 the number of monthly joining members increases again. In June 2022 MongoDB gains 29 new community members,

which is the maximum number in the history of the project. In December 2022, the trend component shows a value of 8. However, throughout the history of MongoDB, usually there are less than 10 monthly joining members.
Corresponding to the small numbers of monthly joining members, the seasonal pattern shows a small amplitude in the beginning of the project. With increasing absolute numbers, the amplitude of the seasonal component increases, in particular visible from 2014. The pattern of the seasonal component remains the same throughout the history of MongoDB. In April, the seasonal component shows a negative value. Every year a peak of monthly joining members is reached in June.
Similarly, the plot of the residual component illustrates small values in the beginning of the project. All in all, we can observe a relatively even distribution of residual values. The greatest absolute values occur in June 2017 with 5.8, May 2019 with 7, and on the negative side in June 2021 with -4.4.
In the months directly before the license change, there is a small decreasing trend of joining members. The license change coincides with the beginning of an increasing trend of monthly joining members which lasts through the next years and until the end of the available data in December 2022. The seasonal pattern remains the same, but as described above the amplitude continues to increase. Regarding the residual component there are small values in the months around the license change. Furthermore, there is no indication of a long term effect.

Figure 7.4 shows the development of individuals leaving the community of MongoDB from the start of the project until 30-06-2022. The figure consists of four plots. The first plot shows the total count of monthly leaving members. Thereafter, three plots show trend, seasonal and residual component according to the STL decomposition. Vertical lines indicate the date of license change, which is 16-10-2018 in the project of MongoDB. Table B2 in appendix B.1.2 contains the data.
The first plot shows only small values in the beginning, in most months no one or only a single member leaves the MongoDB community. Nevertheless, the trend increases as we can observe in the second plot. Subsequently, the trend component remains almost constant from October 2011 to January 2014 with values between 3 and 3.9. The actual number of monthly leaving members varies between 1 and 8 in this period. Starting in 2014 we can observe the next increase of the trend component with values between 4 and 5 from February 2014 until November 2016. In this period the total values of leaving members range from 2 in April 2014 and 13 in August 2016. Then, there is a small decrease in the trend component. In the period of November 2016 to July 2019 the values range from 3.7 in March 2017 to 4.4 in February and March 2018. The total values here vary between no leaving member in September 2018, only one leaving member in December 2016, February 2017, June 2018, February and May 2019, and 14 leaving members in August 2017 which also represents the maximum number of leaving members in the history of MongoDB. After that, we can observe a new increase in the trend from

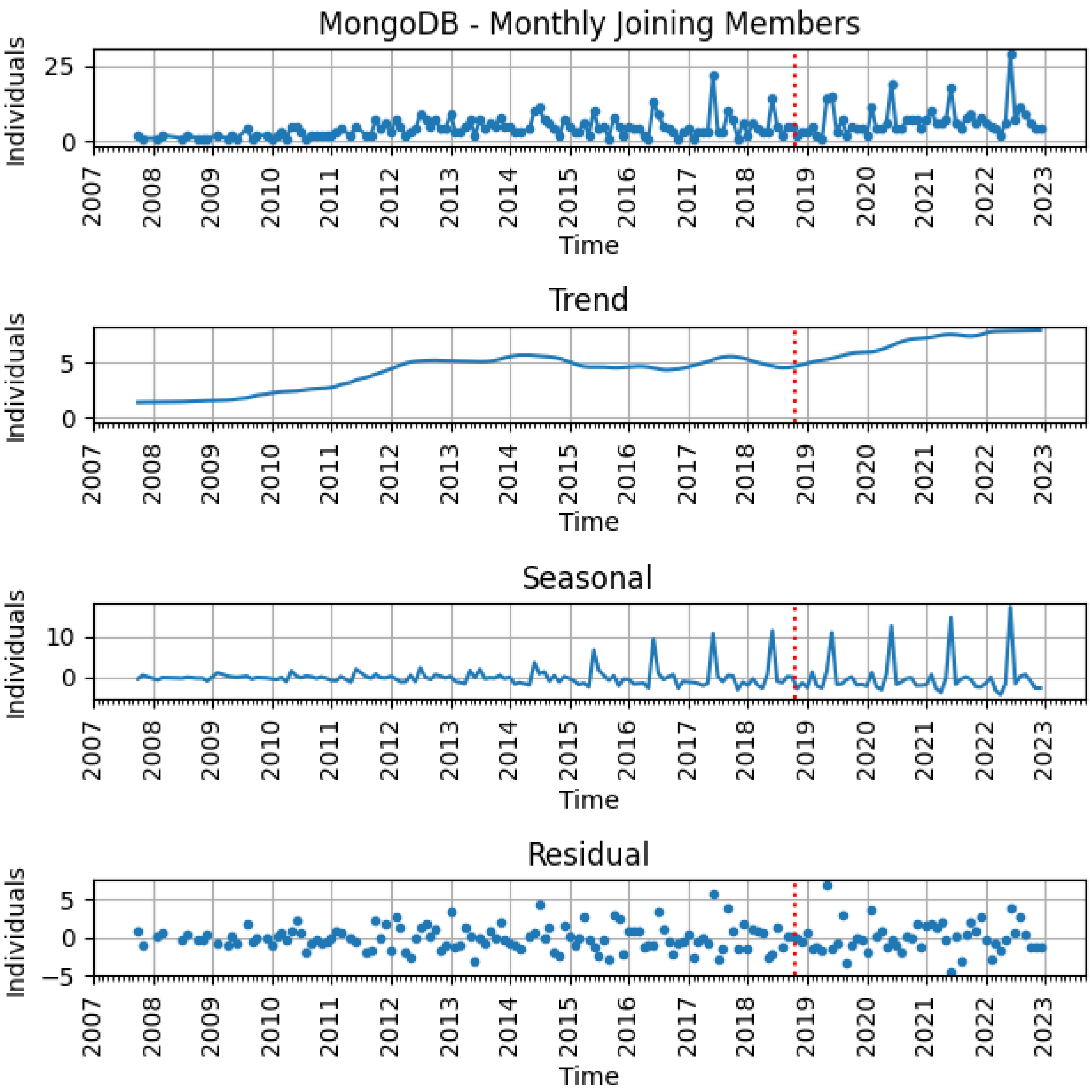

Figure 7.3: STL decomposition of monthly joining community members in the project of MongoDB

August 2019 to the most recent data of June 2022, which shows a trend component of 6.7.

In the seasonal component, there is no visible pattern in the beginning. The absolute values are small with the overwhelming majority of absolute values below 1 with exception of January 2009, November 2009, and May 2010. From 2013 to 2019 we can observe an increasing amplitude and a pattern emerges. This seasonal pattern shows a positive peak in July or August of each year, and negative values in April or May as well as in December or January. In 2019 the pattern changes, and shows now smaller peaks, although this development reverses in 2021.

The residual component in the fourth plot of Figure 7.4 illustrated distributed values throughout the history of MongoDB. In October 2011 a first noticeable residual data point occurs with a value of 3.85. This corresponds to a peak in the absolute value of 8 leaving

members. Furthermore, the two months before October 2011 show 0 leaving members and therefore we can assume that this value represents a catch up. For most of the time, the absolute values of the residual component remain in the same range. Only in the time period from August 2019 to April 2020 we can observe larger residual values. Accordingly, the greatest absolute values occur in October 2019 with -3.99 and in February 2020 with 3.4.
When MongoDB changes its licensing terms, the trend of monthly leaving members was in an almost constant phase. Only a few months later the trend started to increase again. The amplitude of the seasonal component is noticeably smaller after license change and the seasonal pattern fades out in the years following it. Regarding the residual component, we can observe a series of small absolute values around the license change. Notably, before the license change there are five consecutive months of positive residual values, followed by four months of negative residual values. Furthermore, a year after the license change the residual plot shows a period of larger, alternating values, as described before.

Figure 7.5 shows the development of the community of MongoDB according to the onion model. It illustrates the shares of core, regular, and casual members for each quarter from Q3 2010 to Q4 2022. Table B3 in appendix B.1.2 contains the corresponding data.
The share of core members ranges from 42% of all community members in Q4 2014 to 79% in Q1 2021. The first quarter, Q3 2010, shows an exceptional high share of 75%. Subsequently, the proportion of core members lays between 42% and 65% in the period until 2015 Q3. Then, we can observe an increase in the share of core members. Between Q4 2015 and Q3 2022 between 56% and 80% of the community members are core members. In the last quarter shown in Figure 7.5, Q4 2022, core members represent 44% of the community which is noticeably smaller than in the preceding quarters.
The proportion of regular members varies between 0% in Q3 2010 and 42% in Q4 2014. In addition to Q3 2010, the data shows 0% regular members in Q4 2021 as well. In a first period from Q4 2010 until Q3 2015, regular members represent between 20% and 42% of the community. Subsequently, their share decreases. In the period from Q4 2015 to Q4 2022, the proportion of regular members is between 12,5% in Q3 and Q4 2020 and 33% in Q4 2022, with exception Q4 2021 when no regular members are present.
Casual members represent between 6% in Q4 2017 and 25% in Q3 2010 and Q2 2021 of the community. Their share varies slightly but remains around same values.
Shortly before license change there is a local minimum of the share of core members in Q1 2018. Accordingly, before and after the license change the share of core members increases. The proportion of regular members decreases in this period while the share of casual members remains almost constant.

Figure 7.6 shows the development of the share of contributions authored by employees of MongoDB. The share for each month from March 2008 until December 2022 is illustrated. Table B4 in appendix B.1.2 contains the data.

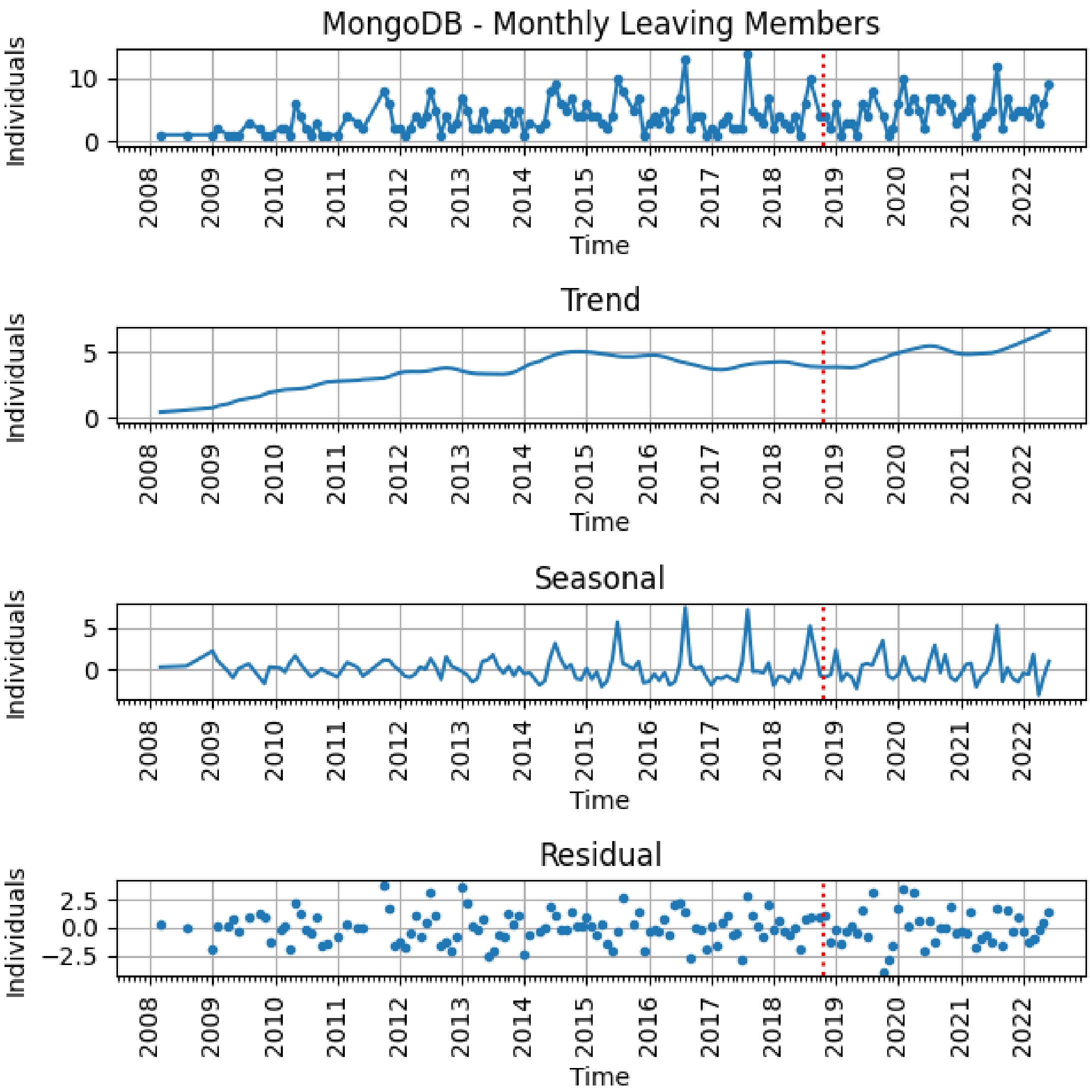

Figure 7.4: STL decomposition of monthly leaving community members in the project of MongoDB

The proportion of contributions authored by MongoDB employees is small in the beginning of the project. In five of the first eight months they even authored 0% of the contributions. Subsequently, their share increases, starting in June 2008 with 14%. At the end of this first period, in December 2008, MongoDB employees authored 30% of all contributions. Then, their share rises sharply. In January 2009 it amounts to 60% and continues to increase until it reaches 94% in April 2009 which is the maximum share in the history of MongoDB.

It follows a period of relatively high shares. From February 2009 to January 2011 between 54%, in April 2010, and 94%, in April 2009, of the contributions are authored by employees of MongoDB. In the following, the share decreases and varies between 20% in September 2012 and 65% in February 2012 in the time period from February 2011 to April 2016. May 2016 initiates an increase with 71%, and in the following time period until December

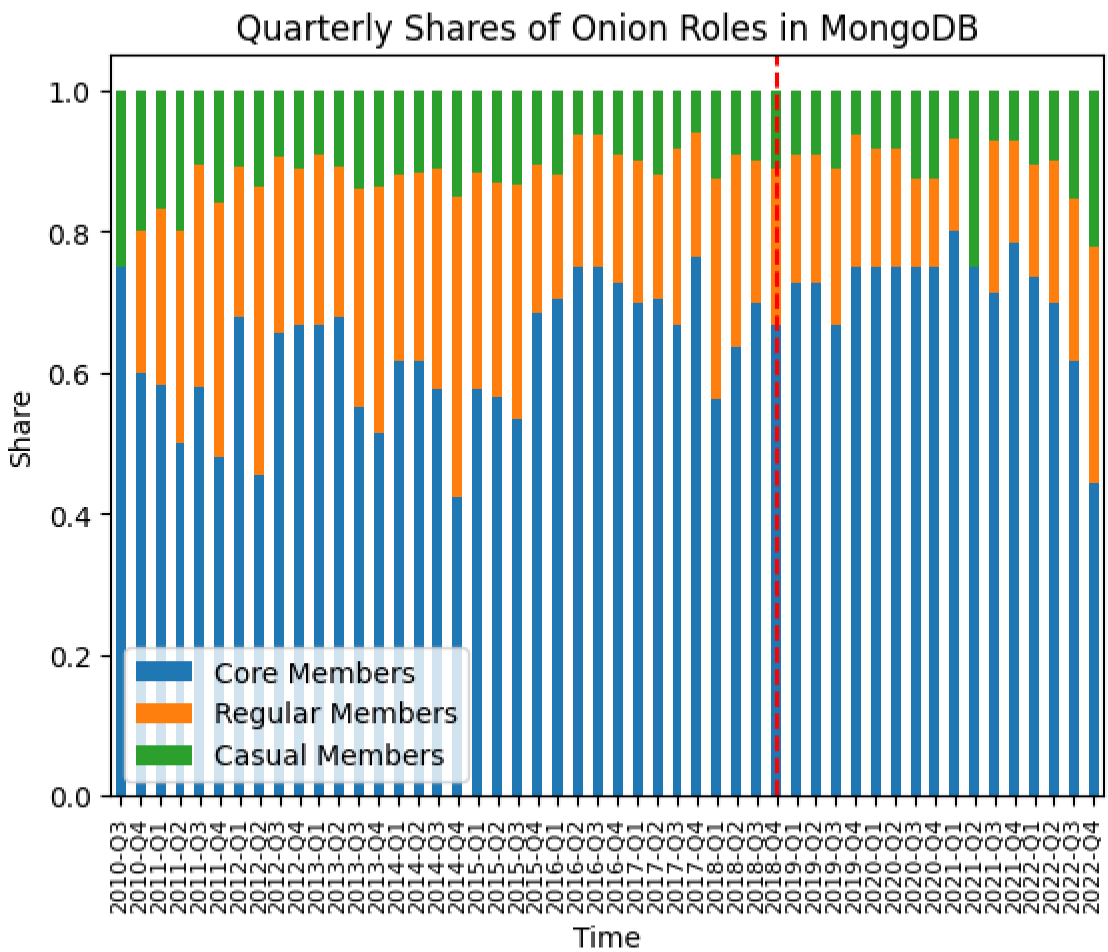

Figure 7.5: Development of the proportions of the onion roles in the community of MongoDB

2022 the proportion remains on a high level ranging from 70% in February 2019 to 91% in December 2019.

The license change occurs in the second period with a high share of contributions by MongoDB. Immediately before or after license change there is no visible sign of an effect. A few months later, we can observe peaks from September 2019 to January 2020. After these months, the share returns to a similar level as before, although we can notice a small increasing trend towards the end of the data.

7.1.3 Summary

The analysis of the activity in the MongoDB project shows differences between technical activity, represented by the number of monthly commits, and social activity, represented by the number of monthly created issues. While the amount of monthly commits remained high, the number of monthly created issues diminished since 2016. Regarding the created issues, this could be a sign of maturation of the project. On the other hand the development

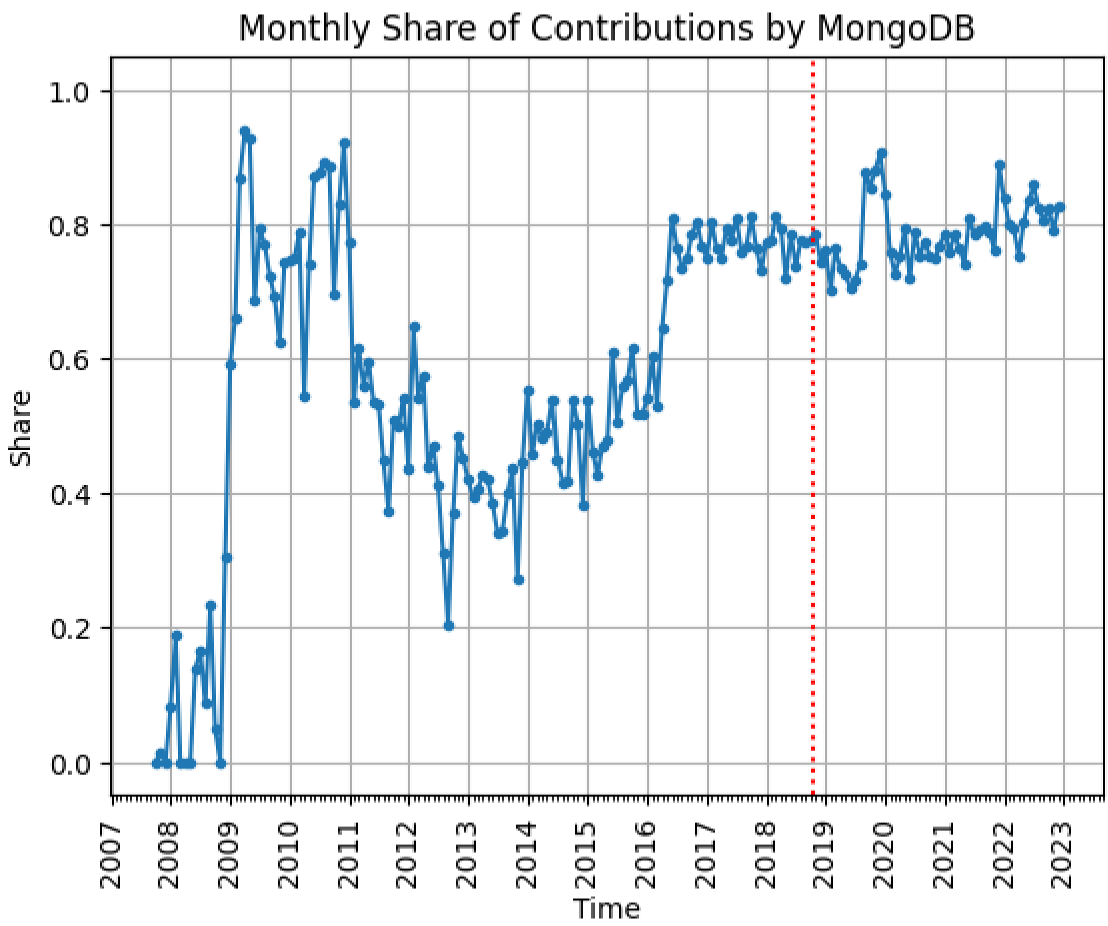

Figure 7.6: Development of the proportion of monthly commits authored by employees of MongoDB

continues as shown in Figure 7.1 which contradicts this speculation.
Figure 7.1 and Figure 7.2 indicate no effect of the license change on the number of monthly commits as well as on the number of monthly created issues.

We observed similar developments in the data regarding monthly joining and leaving members. Both show an increasing trend from the beginning towards the end, with few constant or decreasing periods.
Regarding the onion model, core members represent a large share of the community of MongoDB. Their share increased throughout the history of the project, mostly on the expense of the share of regular members while the proportion of casual members is less affected. However, the last quarter indicates new developments as the share of core members was as small as it was the last time in Q4 2014.
The proportion of contributions authored by employees of MongoDB was very small in the first months of the project. In 2009 it suddenly increased to more than 90%. After a period of smaller values, the share stabilizes from 2016 on at values slightly below 80%.
After the license change, the trends of monthly joining members and monthly leaving

members both increased. Considering the respective seasonal components we could observe opposite developments. While the amplitude of the seasonal component increased regarding monthly joining members, the amplitude decreased in the case of monthly leaving members.
Considering the distribution of onion roles in the community of MongoDB, the license change occurred when the share of core members was in a local minimum. Subsequently, we could observe an increase in the proportion of core members on the expense of the share of regular members.
Regarding the share of contributions by MongoDB employees, the license change occurred when the share stabilized on a high level. There is another peak a few months after the license change, but thereafter the share returns to the previous level.

7.2 Elasticsearch

Here, the results of the analysis regarding Elasticsearch are presented. As described in chapter 6.3, Elasticsearch changed its license on 10-02-2021. The results regarding community activity are discussed first, followed by those concerning the structure of the Elasticsearch community.

7.2.1 Community Activity

Figure 7.7 and Figure 7.8 show the development of activity inside the community of the Elasticsearch project. Figure 7.7 represents the technical activity, measured as commits per month. Figure 7.8 shows the social activity, measured as the number of issues created per month. Table B5 in appendix B.2.1 contains the corresponding data.
Figure 7.7 and Figure 7.8 both display all available data from the creation of the project in February 2010 until 31-12-2022.
In the first three years, 2010, 2011, and 2012, between 69 and 261 commits were contributed each month. The first and second plot of Figure 7.7 display an increase in monthly commits starting in 2013 and continuing until the end of 2015. In 2016 and 2017 the number of monthly commits typically ranges between 900 and 1300, with few exceptions. The trend plot displays this time period as a horizontal line. From 2018 until late 2019, there is another increase in the amount of monthly commits. In the first 10 months of 2019, there are between 1398 and 1896 monthly commits. Subsequently, the number of commits per month decreases, as we can observe in the trend plot. In 2022, the amount of monthly commits ranges only from 423 to 986.
The seasonal component, displayed in the third plot of Figure 7.7, shows changing patterns. In the beginning, the amplitude is small, corresponding to the overall small monthly commits. From 2013 to 2017 there is a seasonal pattern with peaks in the second half of the year. Then, the seasonal component begins to change and from 2018 it shows a drop in monthly commits at the end of the year.

Similarly, the residual component shows a small amplitude in the beginning with values close to 0. From 2014 on there are alternating residual monthly commits which do not exhibit any pattern.
When Elasticsearch changed its license in February 2021, the monthly commits were decreasing. After the license change the decline accelerated as we can observe in the trend plot of Figure 7.7. The seasonal component continues in the same pattern as before. In the residual plot we cannot observe any deviating behavior around or after the license change.

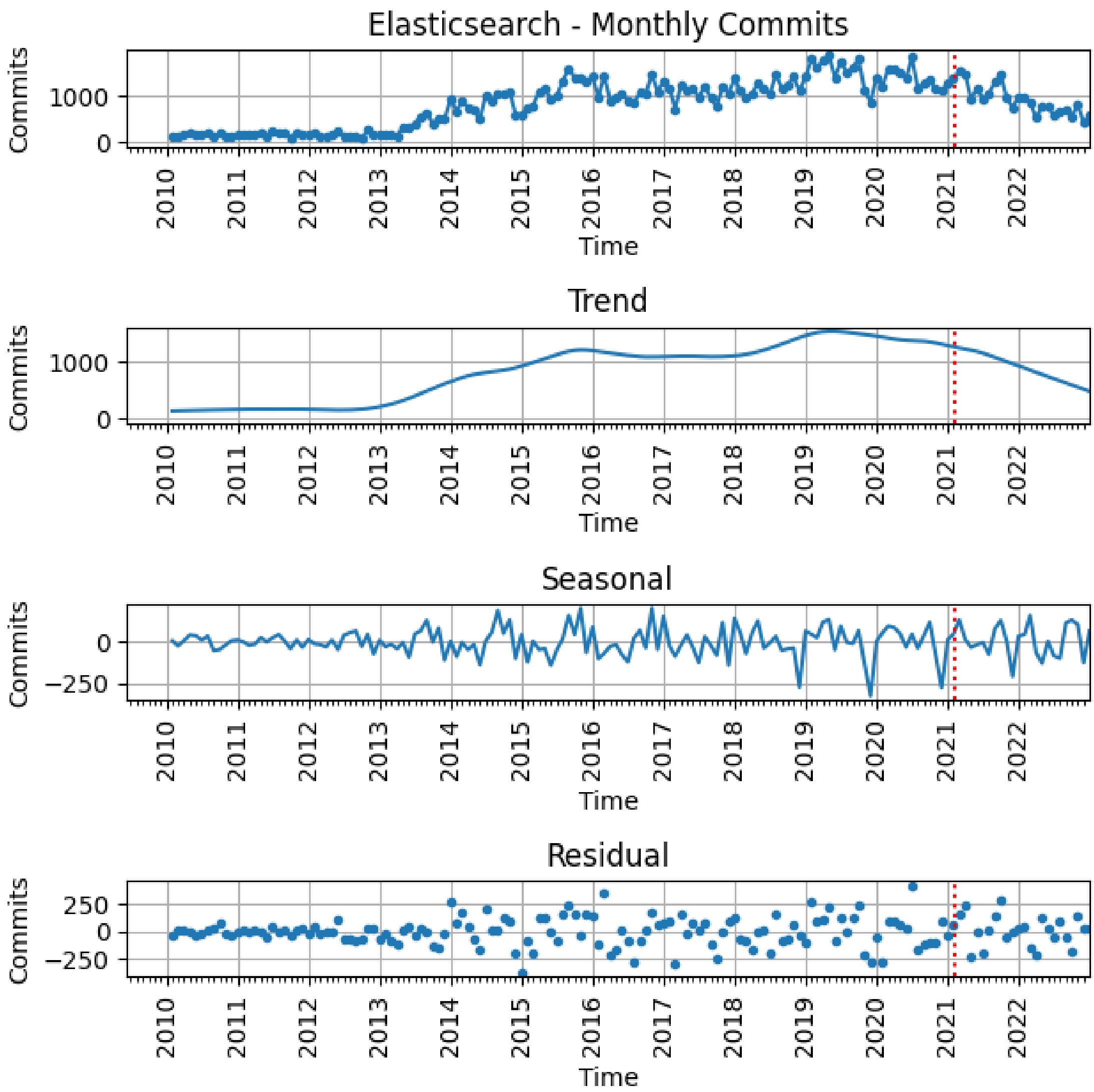

Figure 7.7: STL decomposition of monthly commits in the Elasticsearch project

Figure 7.8 shows the social activity, measured as the number of issues created per month. Table B5 in appendix B.2.1 contains the corresponding data.
Figure 7.8 displays all available data from the creation of the project in February 2010 until 31-12-2022.

In the beginning the number of monthly created issues is relatively small. From February 2010 until February 2013 between 40 and 105 issues are created each month. The 40 new issues in December 2010 represent also the minimum number of monthly created issues. Subsequently, the trend steadily increases until first a maximum is reached in November 2015 with 711 new issues. The value of the trend component in this month is 589. From November 2015 to September 2017, the trend slowly decreases and reaches a minimum with a value of 489. Afterwards, there is another increase of monthly created issues. In May 2018 734 issues are created, which represents the first month with more than 700 created issues since November 2015. The increase is visible in the trend plot and the trend component reaches a in September 2020 with a value of 1358. In absolute numbers, the maximum of monthly created issues occurs in July 2020 with 1763 issues. Subsequently, the absolute numbers and correspondingly the trend decreases. In the last month, December 2022, there are 746 created issues.

The seasonal component, illustrated in the third plot of Figure 7.8, has a small amplitude in the beginning, corresponding to the small absolute number of created issues. Throughout the years the amplitude increases and a pattern emerges. We can observe a negative seasonal component every year in December. Furthermore, most years have a positive peak early in the year, e.g., in March and another peak around August.

Similarly, the values of the residual component are small in the first years. The first notable month is February 2018, where the residual component has a value of -215. A year later, in February 2019 we can see an exceptional large value with 287. Other notable points are -246 in December 2019 and 348 in July 2020, later -199 in May 2021 and 302 in October 2021. All in all, the residual component shows an increased amplitude of values since the end of 2019.

In the project of Elasticsearch, the license change occurred shortly after maximum of monthly issues. The trend already started to decrease and continues to decrease after the license change. The seasonal component shows no indication of an effect of the license change. In the residual plot we can observe that the in months directly around the license change the value of the residual component is relatively small.

7.2.2 Community Structure

Figure 7.9 illustrates the development of monthly joining community members. The figure contains data from the beginning of the project in February 2010 until the end of 2022.

Table B6 in appendix B.2.2 contains the data used to create the figures presented in this chapter.

Until September 2013, the number of monthly joining members remained smaller than 10, with even 0 joining members from March to August 2010. Starting in September 2013, there is an increase as we can observe in the trend plot of Figure 7.9. In May 2014, a first maximum is reached with 30 new community members. The number of monthly joining members remains between 16 and 27 until May 2017, with the exception

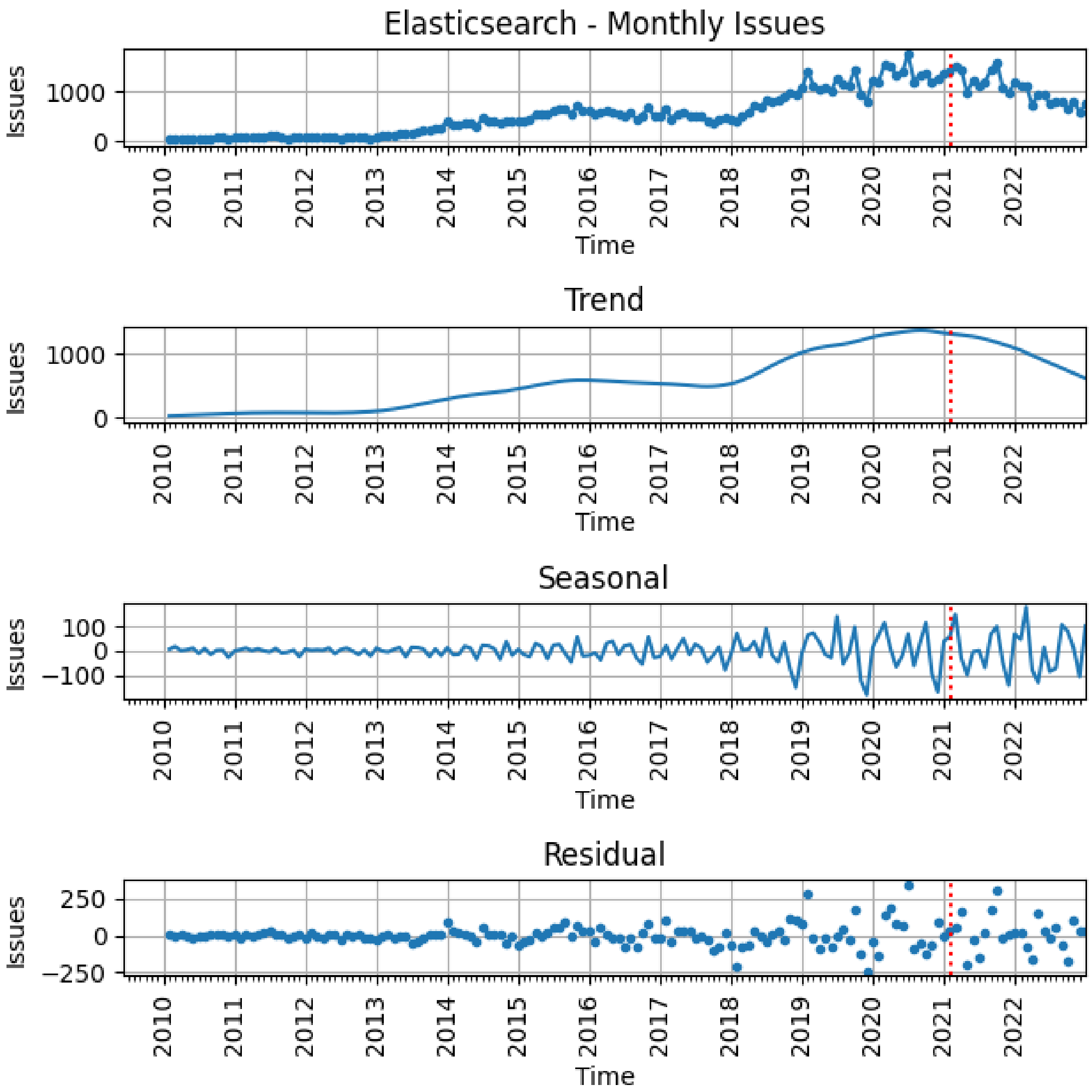

Figure 7.8: STL decomposition of monthly created issues in the project of Elasticsearch

of January 2015, July 2016, and September 2016 with 13, 11, and 9 joining members respectively. In December 2016, the maximum of 30 monthly joining members is reached a second time. From June 2017 until September 2018, there are between 12 and 21 monthly joining members. We can observe this decrease in the trend plot. Starting in late 2018, the number of monthly joining members increases again until the beginning of 2020. Since then, the number of persons joining the Elasticsearch community each month decreased. The trend component, displayed in the third plot of Figure 7.9, indicates steadily decreasing numbers from 19.8 in October 2019 to 9.8 in December 2022.

The seasonal plot of Figure 7.9 shows small alternations in the beginning, but only from 2017 a pattern emerges. Here, we can observe a positive seasonal component in January, and a negative seasonal component in May, June, July, and August.

The residual term shows small absolute values in the time period until the end of 2013.

From September 2014 until October 2015 there is a notable pattern of 4 consecutive months of positive residual monthly joining members, followed by 5 negative months which are in turn followed by 5 positive months. In July and August 2016, the values of the residual term are -8.7 and -3.6 respectively. Subsequently, in September and October 2016 the values are 3.7 and 8.3. In the first plot of Figure 7.9 we can observe a local minimum followed by a maximum in this time period. Therefore, we can interpret the large residual values as a result of this sequence.
Similar phases of larger alternating residual values occur at the end of 2019 and 2020. Thereafter, the residual monthly joining members return to values around 0.
Elasticsearch changed its licensing terms in a period of a decline in monthly joining members. The trend plot indicates a decrease for over a year before the license change. After the license change, the trend continues to decrease, although the decline slows down during the first year after the license change. There are no indications of an impact of the license change on the seasonal or residual component.

Figure 7.10 shows the development of individuals leaving the community of Elasticsearch using data from the start of the project until 31-12-2022. As only those members who made their last contribution at least 6 months ago are considered as members who left the community, there are no leaving members in the last 6 months of 2022.
In the first years there are only few monthly leaving members, as a consequence of the small community. In November 2013, for the first time 10 members left the community. A sharp increase follows, leading to 32 monthly leaving members in May 2014, which represents the maximum number of monthly leaving members in the history of Elasticsearch. In the following three years the number of monthly leaving members varies between 11 and 28 with a slightly increasing trend. The trend component reaches its maximum in December 2015 and remains on a similar level until April 2017. Subsequently, there is a small decline in monthly leaving members. In 2018 each month between 11 and 22 members left the community of Elasticsearch. However, the year 2019 is another phase of increasing monthly leaving members leading to a local maximum at the end of the year as we can observe in the trend plot of Figure 7.10. From January 2020 to January 2021 the trend component decreases steadily from 18.5 to 15.1. Since February 2021 the trend of monthly leaving members stabilizes. This is demonstrated by the values of the trend component in the time period from February 2021 to June 2022, where the trend component varies between 14.5 and 14.8.
The plot of the seasonal component in Figure 7.10 shows a small amplitude in the beginning, corresponding to the small number of monthly leaving members. From 2012 to 2014 we can observe a seasonal pattern with a drop of leaving members at the end of the year. During the next two years the seasonal pattern changes and results in a new pattern emerging in 2017. Here, we can observe two peaks in the first month of each year. Furthermore, the amplitude increased.
The residual component shows values close to zero until the end of 2014, with the

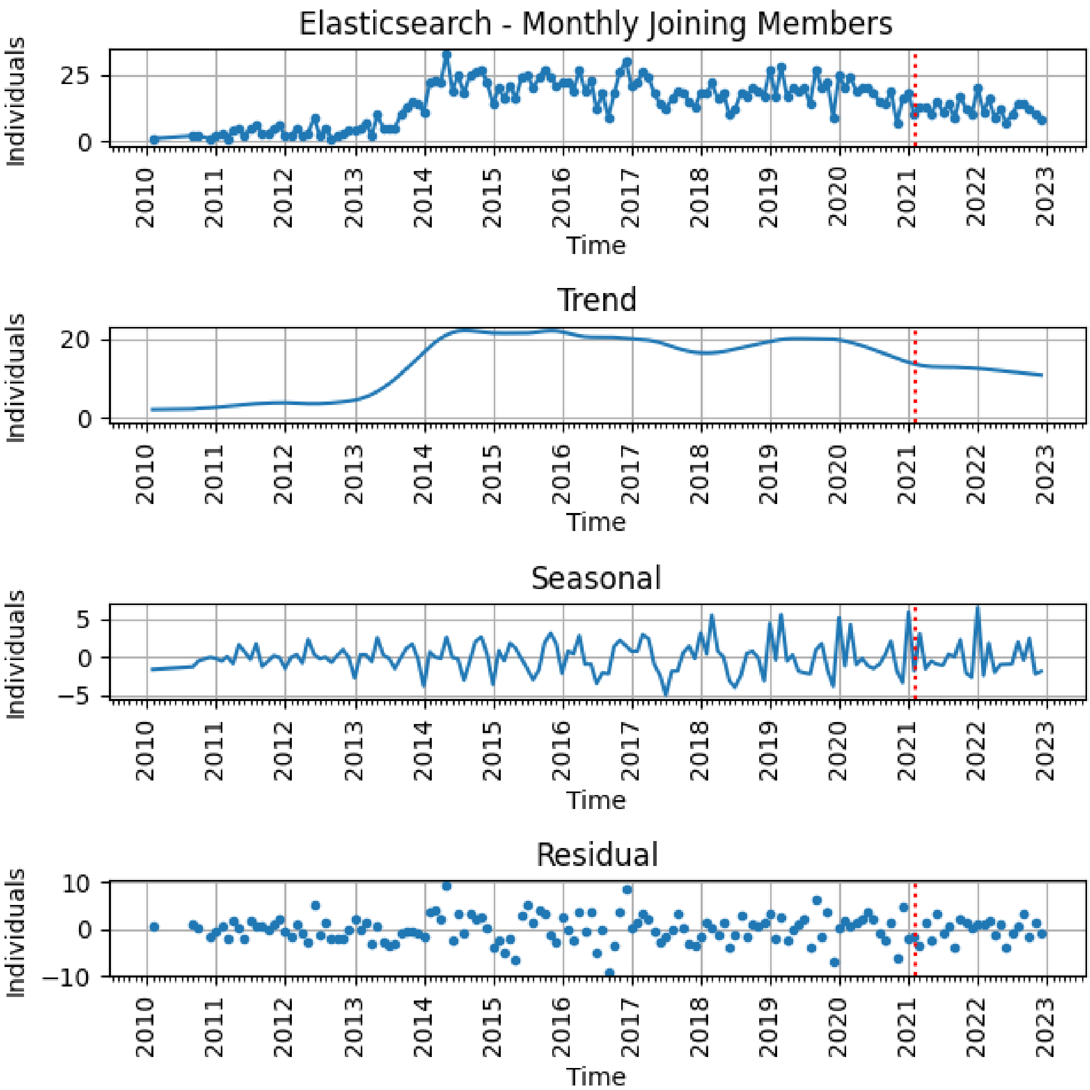

Figure 7.9: STL decomposition of monthly joining community members in the project of Elasticsearch

exception of one outlier in May 2014 which corresponds to the maximum of monthly leaving members in the first plot of Figure 7.10. Starting in 2015 the absolute values of the residual component increases but the distribution does not show any exceptional events. Throughout 2020, there is a decline in monthly leaving members as we can observe in the first plot of Figure 7.10. The end of this decline coincides with the license change on 10-02-2021. After the license change the trend of monthly leaving members continues on a constant level. In the seasonal plot we can observe that the amplitude of the seasonal component increases after the license change, but it continues to follow the same pattern as before. There is no indication of a change visible in the residual component.

Figure 7.11 shows the development of the community of Elasticsearch according to the onion model. Table B7 in appendix B.2.2 contains the data.

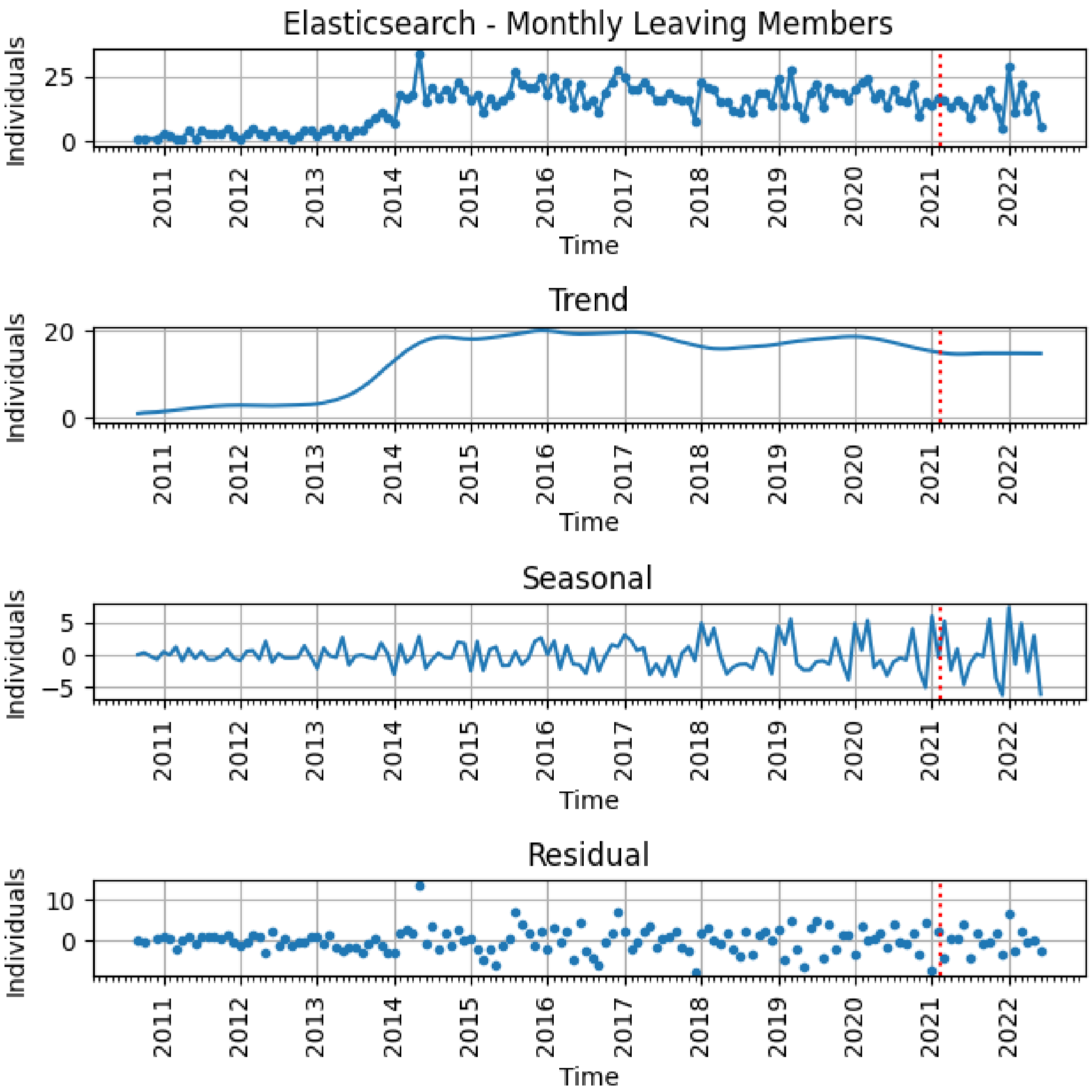

Figure 7.10: STL decomposition of monthly leaving community members in the project of Elasticsearch

It illustrates the shares of the three roles, core members, regular members, and casual members in each quarter from Q1 2010 until the last quarter of 2022.

In the first quarters of the project, the proportion of core members in the community of Elasticsearch is small. With only 5% the minimum share is reached in Q2 2010. Subsequently, the share increases quickly over several quarters to 54% in Q4 2011. Then, it remains relatively stable and from Q4 2011 until Q2 2018 between 43% and 61% of the community members are core members. The 61% in Q4 2012 also represent the maximum share of core members in the history of Elasticsearch. After Q2 2018, the share of core members decreases. In the period from Q4 2018 to Q4 2022 it varies between 35% and 43%.

The proportion of regular members in the Elasticsearch community ranges from 27% in

Q4 2012 to 50% in Q2 2010. In a first period from Q1 2010 to Q3 2011 the share is larger with values between 43% and 50%. Subsequently, it stabilizes between 27% and 39%. Last, casual members represent between 11% in Q2 2013 and 44% in Q1 and Q2 2010 of the community. Similar to regular members, their share is larger in the beginning but decreases from 44% in Q1 2010 to 20% in Q1 2011. Between Q2 2011 and Q2 2018 the proportion of casual members is stable and ranges from 11% to 18%. This period is followed by an increase and from Q4 2018 to Q4 2022 between 24% and 33% of the community members are casual members.
After the license change on 10-02-2021, the distribution of onion roles among the members remained mostly the same. Shortly before and after the license change the shares of the onion roles in the community are stable. In 2022 there is an increase of the share of core members and a small decrease of the shares of regular and casual members.

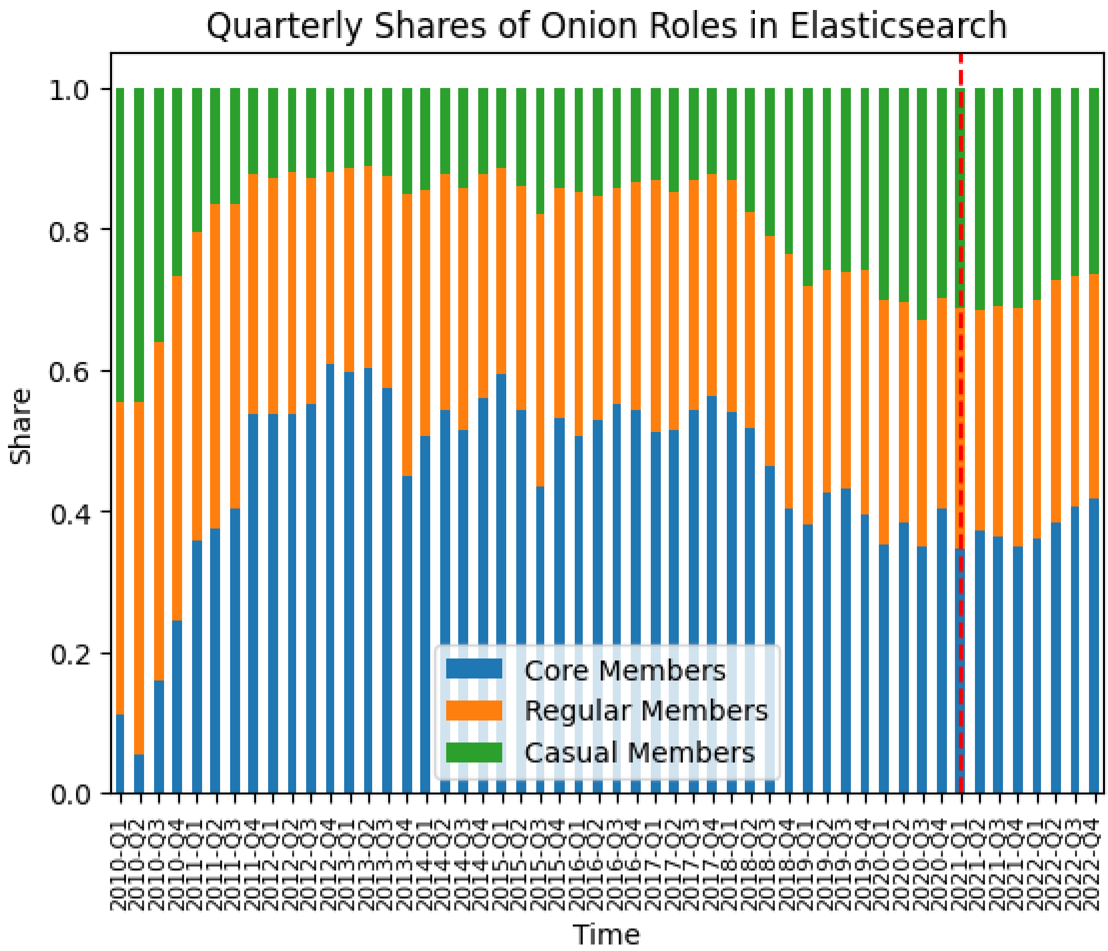

Figure 7.11: Development of the proportions of the onion roles in the community of Elasticsearch

Figure 7.12 shows the development of the share of contributions authored by employees of Elasticsearch. Table B8 in appendix B.2.2 contains the data.

The figure illustrates monthly data from February 2010 until December 2022.

Until August 2012, the proportion of contributions authored by Elasticsearch employees is 0%. The reason for this is that the Elasticsearch project existed before the corresponding company, which was founded in 2012. From 2013 the share increases until it reaches 25% in August 2016. In the period between March 2017 and March 2019 the monthly share remains on a similar level ranging from 25% to 35%. We can observe a sharp increase in the following months leading to the maximum share in the history of Elasticsearch in May 2019 when 49% of all contributions were authored by Elasticsearch employees. In the following months until July 2020, the proportion is always greater than 34%. After July 2020, there is a decrease and in May 2021 a minimum with 25% is reached. Since then, the data shows an increasing trend of the share of contributions by Elasticsearch employees. In December 2022 they authored 30% of the contributions.

The license change occurs in the last decreasing phase, although in the February 2021 there is a small local peak inside the surrounding minimum. Directly after the license change the share decreases for two more months, then it increases.

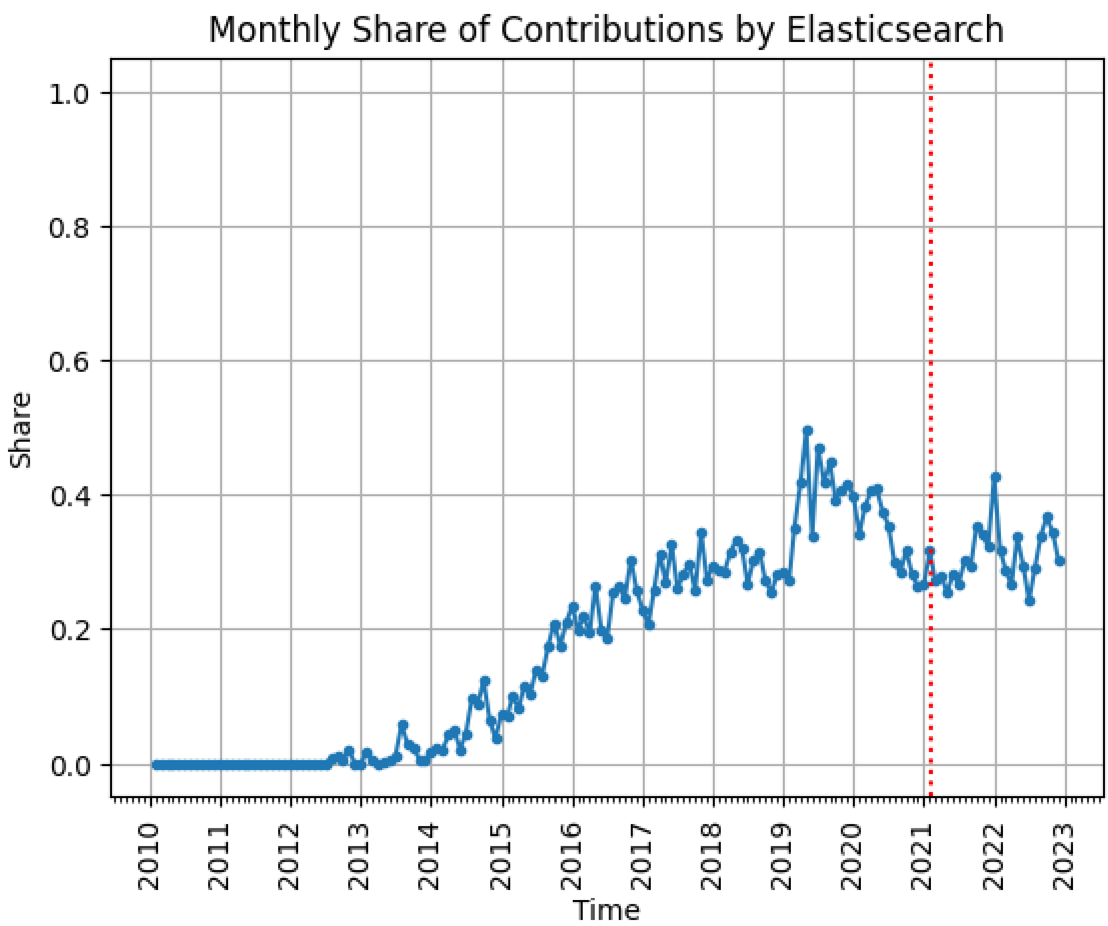

Figure 7.12: Development of the proportion of monthly commits authored by employees of Elasticsearch

7.2.3 Summary

Technical and social activity represented by monthly commits and monthly created issues follow the same pattern in the community of Elasticsearch. In a first period until 2013 the numbers are small. Subsequently, they increase until the end of 2015. Here, they reach a level which they hold until a second phase of increase in 2018. The amount of monthly commits reaches a maximum in 2019, the number of monthly created issues as late as in 2020. Since then, both types of activity diminish. While the seasonal component of the monthly commits shows a change in pattern, the seasonal component of monthly created issues continues in the same pattern but with an increased amplitude.
When Elasticsearch changed its license in February 2021, both the monthly commits and monthly created issues were decreasing. After the license change the decline continues in both cases. Similarly, the seasonal components continues in the same pattern as before. In the residual plots we cannot observe any deviating behavior around or after the license change.

Both monthly joining and monthly leaving members show a similar pattern. In both cases, there are small numbers until 2013. In this year the numbers increase and remain on a higher level in the following years. Towards the end of the investigated period we can observe decreasing numbers of joining and leaving members.
Regarding the roles of community members in the onion model, in the beginning core members represent only a very small share of the community. Subsequently, their share increases and in a first stable period core members represent about 50% of the community, regular members about 35%, and casual members about 15%. In 2018, the share of casual members increases to around 25% on the expense of core members while the proportion of regular members remains stable.
Until August 2012 there are no contributions by employees of Elasticsearch because the company did not exist yet. Between 2012 and 2017 the share of contributions by Elasticsearch employees steadily grows. From 2017 on it stabilizes around 30%, with an interrupting period of higher shares between May 2019 and July 2020.
Elasticsearch changed its licensing terms in a period of a decline in monthly joining and monthly leaving members. However, the decline in joining members continued after the license change while the decline in leaving members stopped with the license change. Instead, the trend of monthly leaving members continued on the level of the month of the license change. The seasonal component of monthly joining members continued unaffected after the license change. Regarding monthly leaving members, the seasonal pattern continued with an increased amplitude. There is no indication of a change visible in the residual components in both cases.
The distribution of onion roles among the members remained mostly the same after the license change apart from a small increase of the share of core members in 2022.
The share of contributions authored by Elasticsearch employees decreased in the two months after the license change before it started to increase again.

7.3 Redis

Here, the results of the analysis regarding Redis are presented. As described in chapter 6.4, Redis changed the licenses of the respective projects twice. The first license change occurred on 15-07-2018, and the second on 21-02-2019. Both dates are highlighted by vertical red lines in the plots. The results regarding community activity are discussed first, followed by those concerning the structure of the Redis communities.

7.3.1 Community Activity

Figure 7.13 and Figure 7.14 show the development of activity inside the community of the Redis projects. Figure 7.13 represents the technical activity, measured as commits per month. Figure 7.14 shows the social activity which is operationalized as the number of issues created per month. Table B5 in appendix B.2.1 contains the corresponding data.
Figure 7.13 and Figure 7.14 both display all available data from the creation of the first project in 2016 until 31-12-2022.
The first plot of Figure 7.13 shows that usually there are between 60 and 300 commits per month in the Redis projects. In 2016, the number of monthly commits was smaller, and with the exception of December below 100 commits. Starting in 2017, the monthly commits increased and were then usually above 100 commits.
This development is also reflected in the trend component displayed the second plot of Figure 7.13. Furthermore, the trend shows a stagnation from 2018 until 2021. Since 2021, the amount of monthly commits increased steadily.
The plot of the seasonal component shows recurring maxima in the middle of the years, followed by a drop in the next month.
In the residual component of the monthly commits of the Redis projects, we can see that there are phases of high alternation from the second half of 2017 to the first half of 2018 and in the middle of 2020. Besides these two phases, the residual component is close to 0. Furthermore, we can notice that phases of several months of consecutive negative residual commits alternate with phases of consecutive positive residual commits. For instance, the second half of 2020 shows negative residual commits, while most of the months of 2021 have positive residual commits.
The trend plot of Figure 7.13 indicates that the first license change occurs while the trend of monthly commits is decreasing. In the following months it reaches a minimum, and by the date of the second license change it is slightly increasing. However, at the end of 2020 the trend starts decreasing again. Furthermore, as the change in the trend is rather small, this cannot be seen as a significant impact of the license changes.
Similarly, the seasonal component does not indicate any change in its pattern around the license changes.
In the fourth plot of Figure 7.13, we can see that the residual component shortly before the first license change until shortly after the second license change was exclusively negative with the exception of January 2019. On the other hand, there are other periods with

consecutive negative residual commits, as described above. Therefore, we cannot see this period as an indication of impact of the license change on the monthly commits.

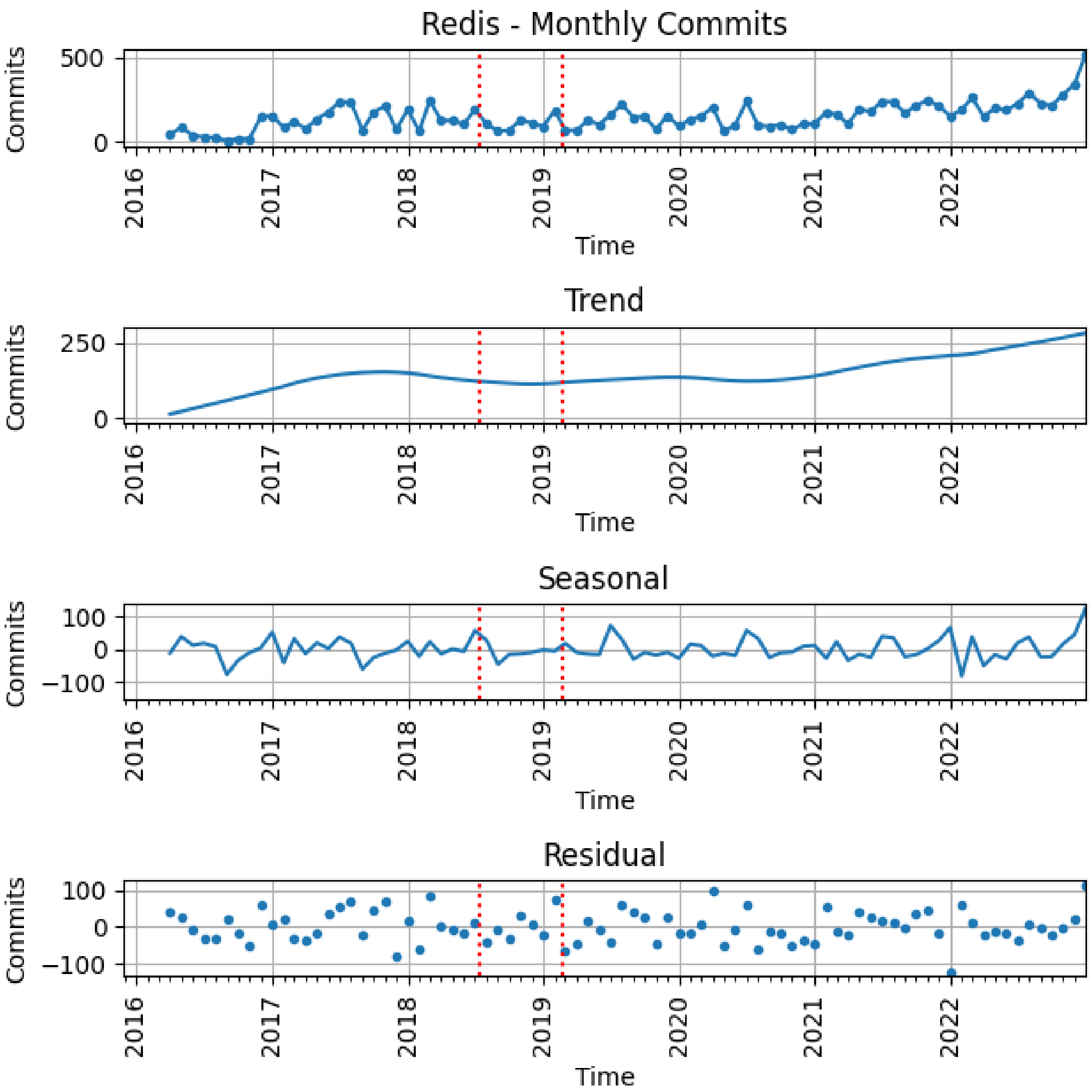

Figure 7.13: STL decomposition of monthly commits in the Redis projects

Figure 7.14 displays the STL decomposition of the monthly created issues in the Redis projects.
The first plot shows that until June 2018 the amount of monthly created issues remained under 50. Since then it increased and apart from few exceptions each month between 100 and 200 issues are created in the projects.
Correspondingly, the trend plot shows a steadily increasing trend line.
In the third plot of Figure 7.14, which displays the seasonal component, we can see a repeating pattern. Every year, the most issues are created in a month in the middle of the year. At the end of each year there is a drop of created issues.

The residual component can be divided into five phases. In the first phase from the beginning of the project until July 2018, the residual created issues are almost 0. Then, there is a short phase of alternating residual issues until December 2019. Subsequently, the residual component returns to values around 0 from December 2019 to December 2020. The fourth phase spreads throughout 2020 and 2021, where we again can observe high alternation of residual created issues. In 2022, the residual component stabilizes again and displays values close to 0.
The first three plots of Figure 7.14 do not show any indications of an impact of the license changes on the monthly created issues in the Redis projects.
However, in the residual plot we can observe a phase of alternation directly after the first license change. This could represent a sign of an impact of the license change, especially since the values of the residual component were close to 0 in the two years before and in the year after the license changes. On the other hand, there is another phase of alternating residual issues in 2020 and 2021. Remarkably, this phase lasts for two years. Combining these insights, we could assume a short phase of confusion after the first license change.

7.3.2 Community Structure

Table B10 in appendix B.3.2 contains the data used to create the figures presented in this chapter.
Figure 7.15 shows the development of joining community members per month in all Redis projects from the beginning of the first project in April 2016 to the end of 2022.
Throughout the whole history of the projects, the number of monthly joining members remains small. The months showing the greatest values are December 2020 and February 2021 with 8 new members respectively. Often, there is only a single person joining the community and sometimes even no one.
In the trend plot of Figure 7.15 we can observe a horizontal line in 2016 and 2017, indicating a constant average of monthly joining members. Since 2018 there is an increasing trend until the end of the analyzed time period.
The seasonal component only shows a pattern from 2020 on, but its absolute value remains small.
In the residual plot, the values in the beginning are close to 0. From the second half of 2020, the values of the residual monthly joining members increases and shows more alternation. After march 2022, the residual values return back to small values.
At the time of the first license change, the trend of monthly joining members was increasing. Subsequently, the increase slows down and changes to a horizontal line at the time of the second license change. In the following years, the number of monthly joining members continued to increase. However, the total numbers are so small that these observations do not allow to draw any conclusions.
The seasonal component shows no indication of an impact of the license changes. In the residual plot of Figure 7.15 we can observe a peak shortly before the first license change,

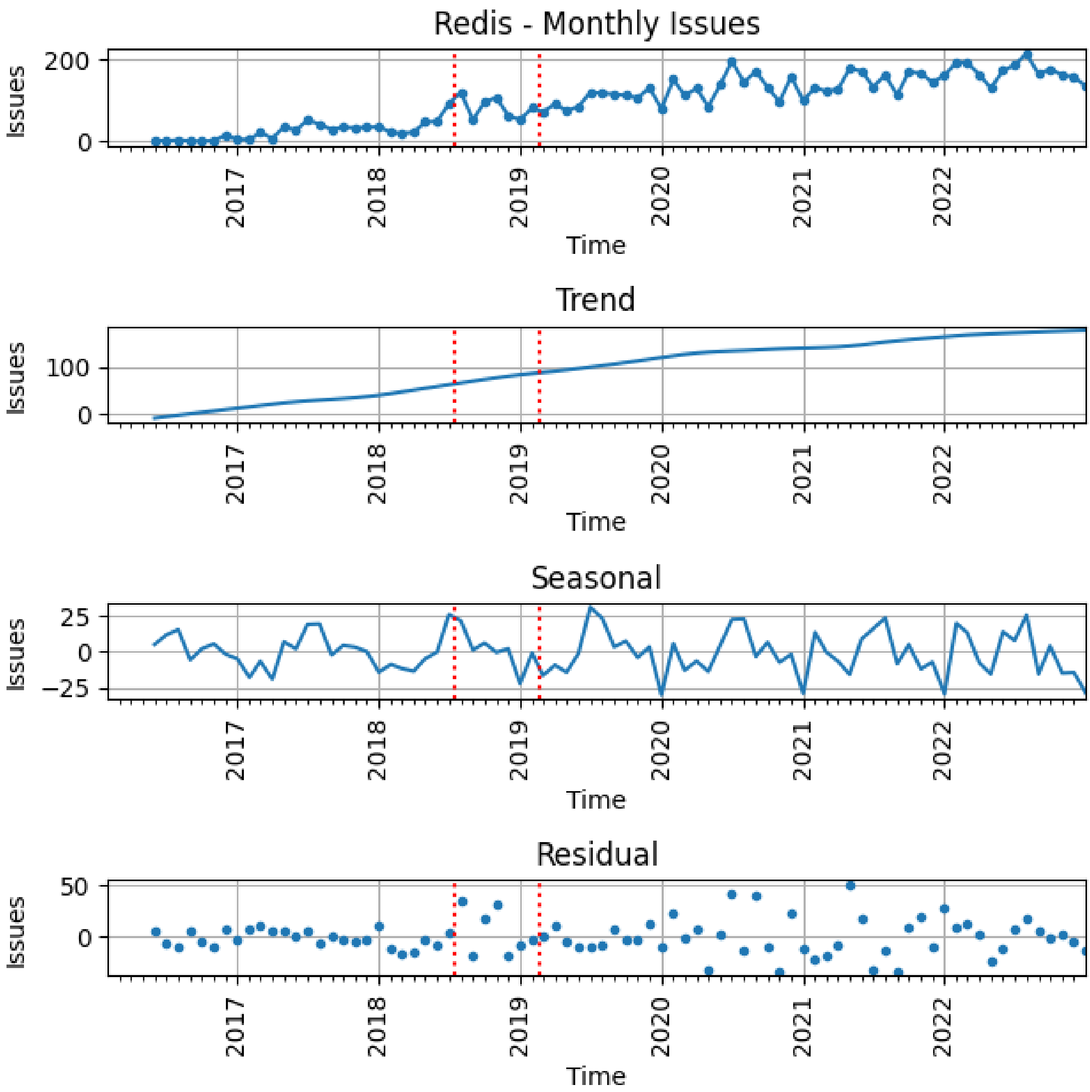

Figure 7.14: STL decomposition of monthly created issues in the projects of Redis

as well as in between the two license changes. Again, the absolute values are small.

Figure 7.16 shows the development of individuals leaving the communities of Redis using data from the start of the first project in April 2016 until 31-12-2022. As only those members who made their last contribution at least 6 months ago are considered as members who left the community, there are no leaving members in the last 6 months of 2022.
Throughout the whole history of the projects, the number of monthly leaving members remains small. May 2018 is the first month with at least 5 leaving members. Later peaks occur in June and December 2020 with 7 and 8 leaving members respectively.
The trend plot of Figure 7.16 shows a slowly increasing graph until the end of 2018. Subsequently, there is a small decrease in the first half of 2019 followed by an increase until the end of 2020. From 2021 on, the trend decreases. However, the differences between maximum (3.5) and minimum (1.7) of the trend component are small because of

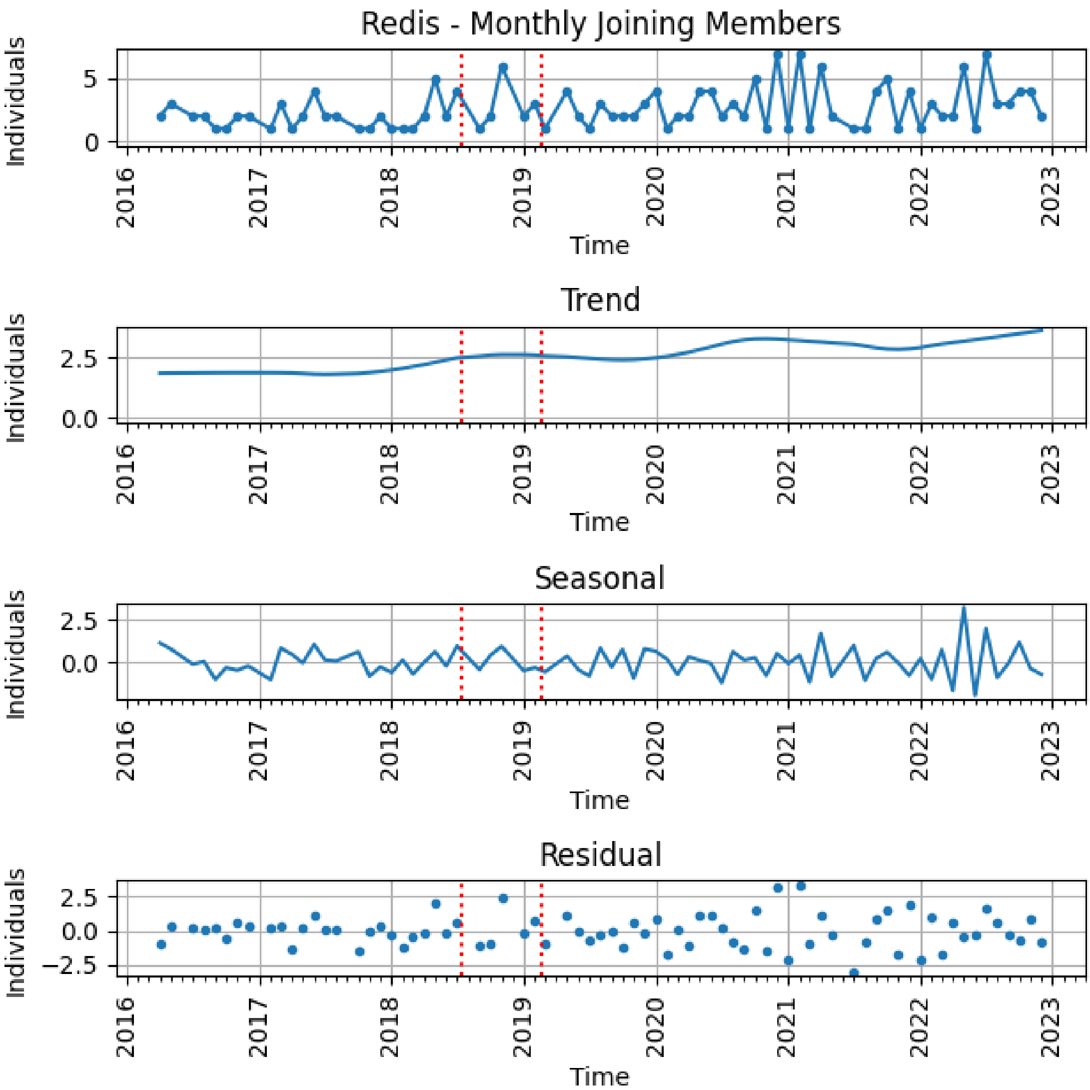

Figure 7.15: STL decomposition of monthly joining members in the communities of Redis

the small number of monthly leaving members.

Regarding the seasonal component, there is no pattern we can identify, but the absolute values increase towards the end of the time period.

The residual component shows short periods of alternating, comparatively larger values and longer periods of small residual values. Throughout 2016 and 2017, the absolute values are small. In the first half of 2018 we observe a short period of alternating values with a peak in May with a residual value of 2.3. Then, the residual values are very small until the next period of alternating values in May, June, and July 2020. Four months of small values follow before the last period of larger, alternating residual values occurs from December 2020 to April 2021.

In the first plot of Figure 7.16 we can observe a peak three months before Redis changed the licenses of the projects for the first time. Another, smaller peak occurs in between the license changes in August 2019. Regarding the trend plot, the first license change coincides with the start of a period of a decreasing trend. A year later, after the second license change, the trend increases again. The seasonal plot shows no indication of an effect of the license changes. In the residual plot we can observe that the first phase of larger, alternating values occurs before and ends with the first license change.

However, we must consider that the absolute number of monthly leaving members is small. Consequently, the peaks before and between the license changes represent 5 and 4 leaving members instead of the usual 1 to 3 monthly leaving members in that period. Therefore, they do not allow to draw meaningful conclusions.

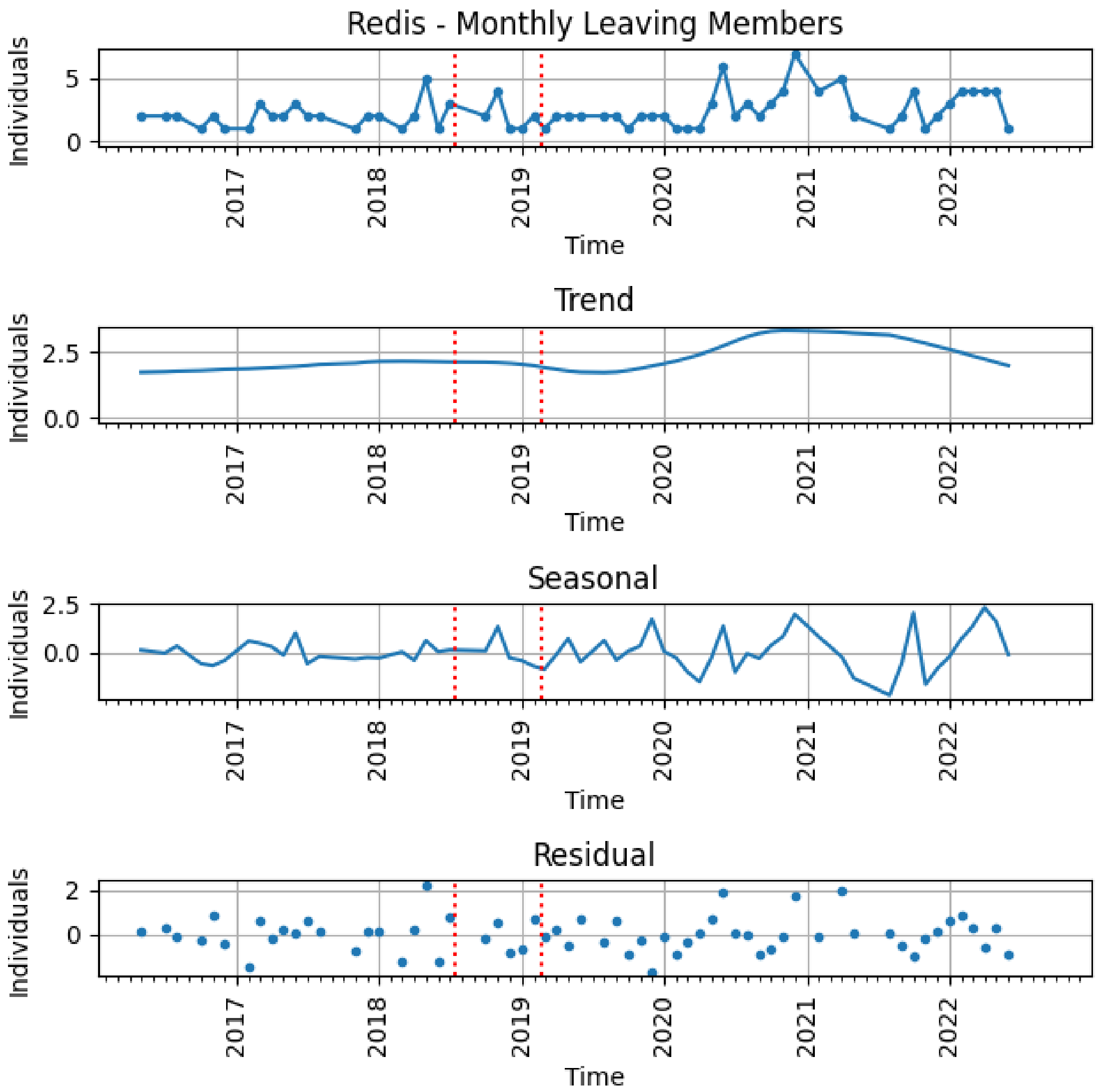

Figure 7.16: STL decomposition of monthly leaving members in the projects of Redis

Figure 7.17 shows the development of the community of Redis according to the onion model. Table B11 in appendix B.3.2 contains the data.

It illustrates the shares of the three roles, core members, regular members, and casual members in each quarter from Q2 2016 until the last quarter of 2022.
The proportion of core members in the communities of Redis varies between 0% in Q2 2016 and 54% in Q1 2018. In the first two quarters, core members represent only a small share of the community. In the following their share increases sharply from 0% in Q2 2016 to 47% in Q1 2017. Subsequently, their share is relatively stable with values between 39% in Q2 2017 and 47% in Q1 2017.
Regular members represent between 14%, in Q2 2016, and 33%, in Q3 2016, of the community. Their smallest share occurs in Q2 2016, directly followed by the maximum share in Q3 2016. In the remaining quarters, their proportion remains between 20% in Q4 2016 and Q1 2017 and 33% in Q4 2020.
The share of casual members ranges from 23% in Q1 2022 to 86% in Q2 2016. Corresponding to the small shares of the previous two roles, we can observe large values in the beginning, but the share of casual members quickly decreases. In the time period between Q1 2017 and Q4 2022, the proportion remains stable with values between 23% in Q1 2022 and 33% in Q1 and Q2 2017.
When Redis changed the licenses for the first time, the share of core members is in a local minimum. In the following quarters it increases, throughout the second license change in Q1 2019. During this period, the proportions of regular and casual members decrease slightly. Subsequently, we can observe the opposite development and there is no indication of long term change visible.

Figure 7.18 illustrates the share of contributions authored by employees of Redis for each month from April 2016 until the end of 2022. Table B12 in appendix B.3.2 contains the corresponding data.
In the period from April 2016 until February 2017, employees of Redis only contribute to the project in May 2016 when they authored 5% of the contributions. From March to August 2017, there are contributions by employees of Redis, in particular in May 2017 when their contributions represent 38% of all contributions which is the maximum share in the history of the investigated Redis projects. After August 2017 there are again no contributions by employees of Redis until February 2018. In May 2018 next peak occurs with a share of 26%. Subsequently, the share decreases but does not return to zero with the exception of September 2018. Instead, the proportion ranges from 1% to 6% in the period between July 2018 and May 2019. Then, the share increases and between June 2019 and September 2020 it varies between 3% and 17%. From October 2020 until February 2021 we can observe a small local minimum. Since May 2021, the proportion of contributions authored by employees of Redis shows an increasing trend, including another peak in September 2021 with a value of 31%.
Right before the first license change the second peak occurs in May 2018 when the share of contributions was 26%. After the first license change smaller values follow. Only after the second license change the share increases again.

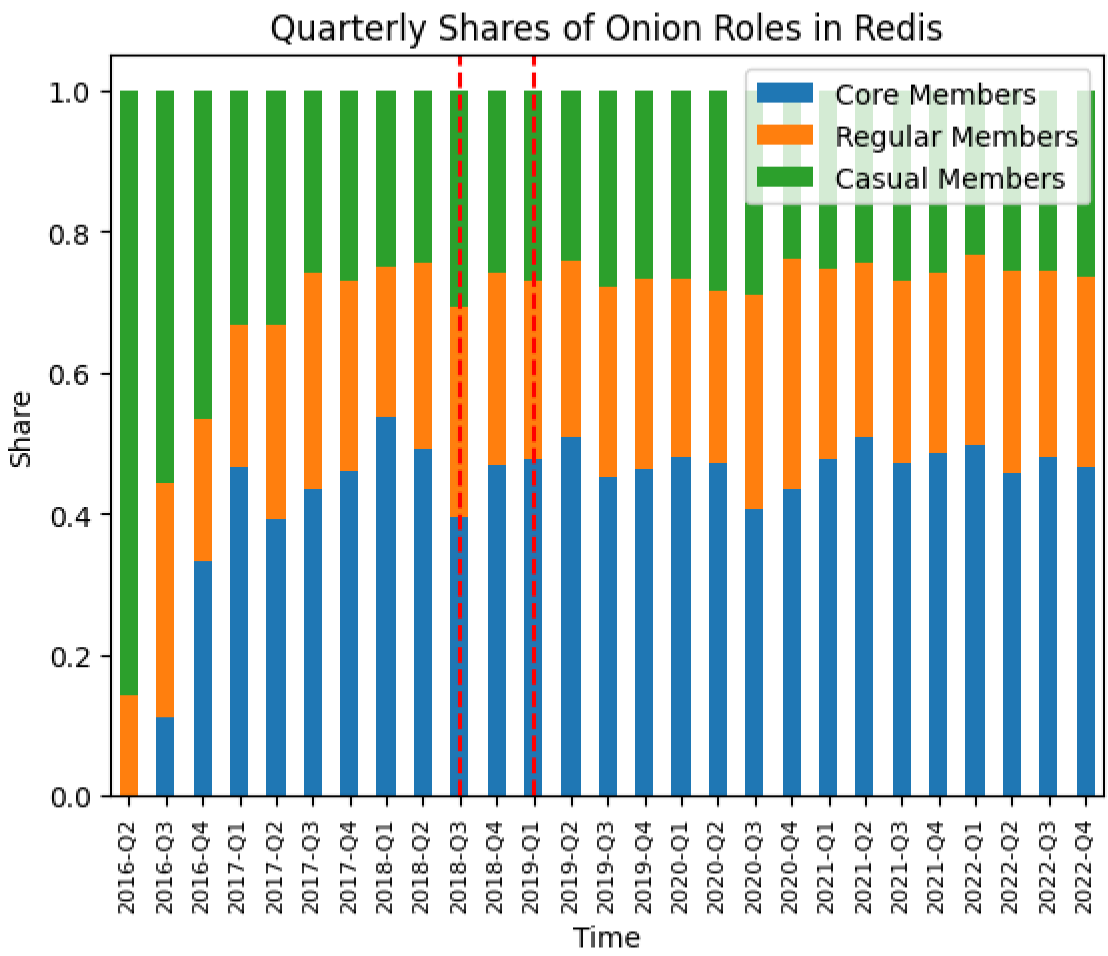

Figure 7.17: Development of the proportions of the onion roles in the community of Redis

7.3.3 Summary

The analysis of community activity in the Redis projects shows an increased activity in both aspects, technical activity and social activity. Furthermore, monthly commits and monthly created issues both have a seasonal component without significant pattern changes over the years. Regarding the residual components, in both cases we can identify phases of values close to 0 and phases with alternating values. However, the phases do not occur at the same points in time, or in any order that could indicate a relation between them.
Figure 7.13 and Figure 7.14 indicate no effect of the license change on the number of monthly commits as well as on the number of monthly created issues in the Redis projects. The only exception is the short phase of high alternation in the residual component of the monthly created issues in Figure 7.14.

In the communities of Redis, the number of monthly joining members shows an increasing trend throughout the history of the projects. Regarding the number of monthly leaving members, the numbers develop differently. Here, they increase slower until the end of 2018. Then, we can observe a small minimum followed by a maximum of monthly leaving

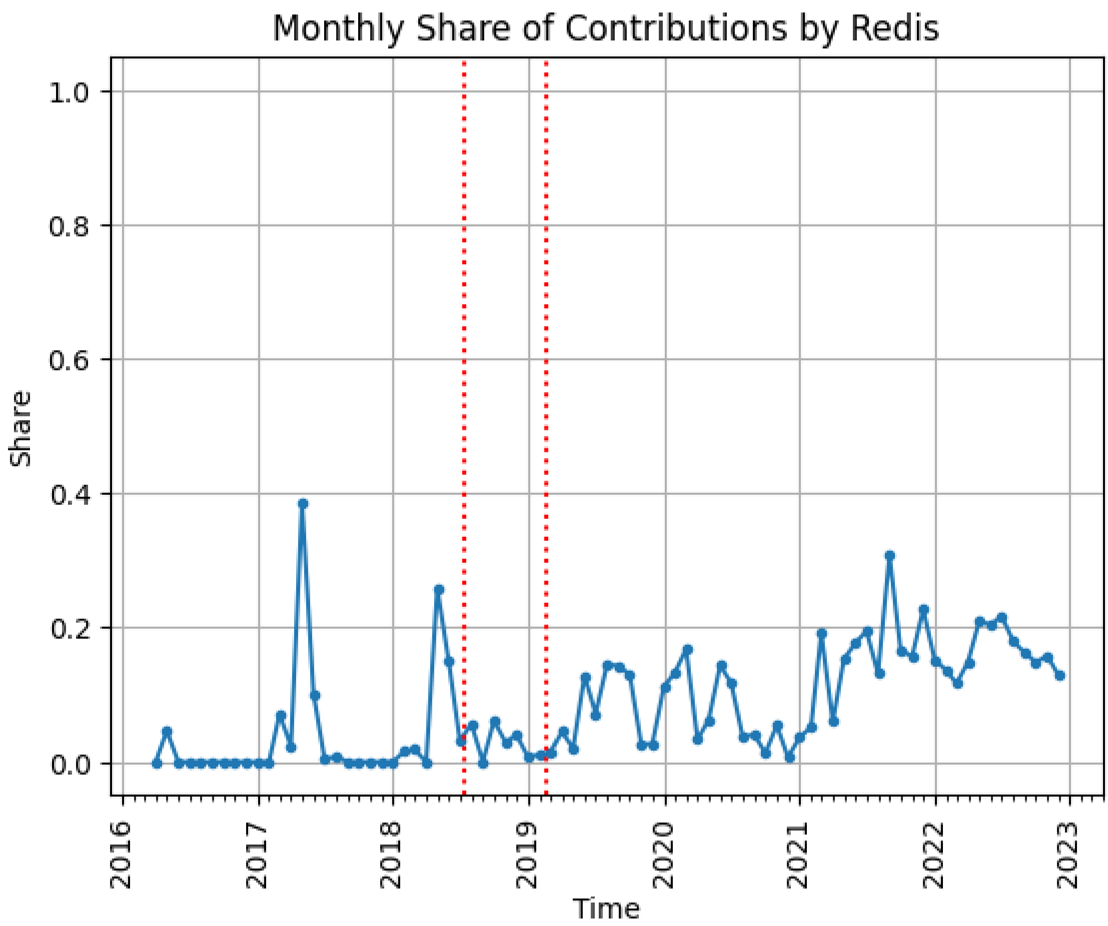

Figure 7.18: Development of the proportion of monthly commits authored by employees of Redis

members. Most notably, the trend is decreasing at the end of the investigated time period. When analyzing the onion roles in the communities, we could observe an initial phase with changing shares until Q1 2017. Subsequently, the shares varied only little.

In the first two years periods of no contributions by employees of Redis alternate with peaks of high proportions. Only starting in July 2018 we can observe a slowly increasing trend of the share of contributions authored by employees of Redis.

In case of monthly joining members the first license change occurs in a period of increasing trend, while it marks the beginning of a period of decreasing trend in the case of monthly leaving members. In both cases the trend then is accelerated in negative direction, leading to a constant trend regarding joining members and a proceeding decreasing trend regarding leaving members throughout the date of the second license change. Additionally, in both cases the trend resumes to increase about half a year after the second license change. The seasonal component shows no indication of an impact of the license changes in both cases.

Considering the onion model, we could observe an increase in the share of core members between the license changes. However, this increase is small and reversed in later

quarters.
Regarding the share of contributions authored by employees of Redis, the first license change occurs shortly after a peak in May 2018. Between the two license changes, the share is smaller, and only after the second license change it increases again.

7.4 Examination of Propositions

In this section the results of the projects are compared. The resulting insights are then evaluated against the propositions proposed in chapter 4.

7.4.1 Community Activity

When comparing the community activity in the three companies, we observe the greatest absolute values in the community of Elasticsearch. This is the case in respect to the number of monthly commits, but even more in respect to the number of monthly created issues. In the community of Elasticsearch there are many months with more than 1000 commits or created issues. Regarding the monthly commits in the community of MongoDB, such numbers are rarely reached. In comparison to the communities of Redis, where usually the number of monthly commits is below 300, the difference is even more notable. Considering the monthly created issues we observe that the numbers in the community of MongoDB are unusually small. While MongoDB has a lot more monthly commits than Redis, the number of monthly created issues is smaller.
In the beginning of the projects, the technical and social activity in the projects of MongoDB and Redis both directly increase. In contrast, the Elasticsearch project experiences a longer period with small number of commits and issues in the beginning before a notable increase starts. This could be related to the fact that the company behind Elasticsearch was founded only in 2012.
The development of monthly created issues in the community of MongoDB exposes the most noteworthy plot with declining numbers while simultaneously the number of monthly commits continues to rise.
At the end of the inspected time period, the monthly commits and created issues of Elasticsearch decrease. In case of Redis, both continue to increase. In the community of MongoDB, the commits decline but the number of monthly created issues increases.
Considering the seasonal components of the monthly commits, we can observe changing seasonal patterns in the project of Elasticsearch, while MongoDB shows only an increased amplitude and Redis a constant seasonal component. Regarding the monthly created issues, we can observe an increased amplitude in the community of Elasticsearch, and changing patterns in the MongoDB project. The seasonal component of monthly created issues in the projects of Redis remains constant.
To summarize the most important observations, in the project of Elasticsearch technical and social activity follow a similar trend. In the projects of Redis, both types of activity

follow the same direction but the technical activity is constant for 4 years in between, while the social activity continuously increases. The most noteworthy development is exposed by MongoDB where technical activity continued on a high level throughout the history but social activity diminished in 2014.

We conducted this part of the analysis to examine the first proposition proposed in chapter 4.1.

> *Proposition 1: Changing the license from an open source license to a cloud protection license leads to reduced community activity.*

In the project of MongoDB, the data shows no indications of an impact of the license change on technical or social activity. Technical activity, operationalized as monthly commits, continued to increase, while social activity, measured in monthly created issues, continued to decrease. We can observe this in Figure 7.1 and Figure 7.2.
Similarly, in the projects of Redis the data does not show any long term effect of both license changes on technical and social activity, as we can see in Figure 7.13 and Figure 7.14.
In the project of Elasticsearch, both types of activities decline after license change, but in both cases the trend already started months before the change. This can be observed in Figure 7.7 and Figure 7.8.
In all projects, the seasonal components are not affected, and in the residual components there is no short term reaction visible. The only exception is a phase of high alternation of the values of the residual component between the two license changes in the projects of Redis. However, there is no straightforward interpretation of this observation.
In conclusion, the increasing technical activity in the project of MongoDB and the increasing social activity in the projects of Redis directly contradict the proposed proposition. Furthermore, the decline of technical activity in the project of Elasticsearch and of social activity in the projects of Elasticsearch and MongoDB, which coincide with the corresponding license changes, are present months before the license changes. Therefore, they cannot serve as support for proposition 1. On the contrary, they demonstrate the absence of an impact, together with the unaffected technical activity in the communities of Redis.

7.4.2 Community Structure

Considering monthly joining and leaving members, Elasticsearch exposes the largest numbers among the projects, corresponding to the size of its community. In the community of MongoDB, notably less individuals join or leave each month, and in the projects of Redis, the numbers are even smaller.
Again, we can observe very small numbers of joining and leaving members in the beginning of the Elasticsearch project. We can notice that the trend of joining and leaving members develops similar in each of the respective projects. Nevertheless, there are periods when the trends progress differently. For instance, at the end of the analyzed period of time,

we observe a decline in joining members but constant numbers of leaving members in the Elasticsearch community. In the communities of Redis, the number of monthly joining members increases at the end but the number of leaving members declines.
The trend of joining and leaving members increases after the license change of MongoDB. In the project of Elasticsearch, the trend of joining members continued to decline after the license change, while the trend of leaving members stopped to decline. Regarding the projects of Redis, we could observe that the license changes coincide with a negative development of the trends of joining and leaving members, resulting in a constant trend for the first and a decreasing trend for the latter.
The seasonal component of monthly joining members shows an increased amplitude in the project of MongoDB and is unaffected in the projects of Elasticsearch and Redis. Considering the seasonal component of monthly leaving members, we could observe a decreasing amplitude in the community of MongoDB, an increasing amplitude in the project of Elasticsearch, and an unaffected seasonal component in the communities of Redis. In all projects, the residual component did not indicate any impact.
We conducted this analysis to examine proposition 2 and proposition 3.

> *Proposition 2: Changing the license from an open source license to a cloud protection license leads to less persons joining the community.*
>
> *Proposition 3: Changing the license from an open source license to a cloud protection license leads to more persons leaving the community.*

Regarding the evaluation of both propositions, the three projects investigated show different developments.
The data of monthly joining members in the communities of MongoDB and Redis contradicts proposition 2 because it shows increasing numbers of joining persons, as we can observe in Figure 7.3 and Figure 7.15 respectively. For the community of Elasticsearch, Figure 7.9 displays a decrease in joining members but the trend was already declining before the license change.
The insights about the development of monthly leaving members in the community of MongoDB is congruent with proposition 3, as we can see in Figure 7.4. Considering the community of Elasticsearch, there is no increase in monthly leaving members after the license change, but the decline stops which shows a development in positive direction. We can observe this development in Figure 7.10. In the communities of Redis, the license changes coincide with a development of the trend in negative direction (see Figure 7.16), contradicting proposition 3 as well.
Therefore, the evaluated data suggests no support for proposition 2 and proposition 3.

Each of the investigated project exposes a characteristic distribution of roles according to the onion model. Notably, in MongoDB the share of core members is significantly larger compared to Elasticsearch and Redis.

Redis and Elasticsearch both show small shares of core members in the very beginning, followed by a sharp increase and a stable period. MongoDB on the other hand starts without regular members at all and an overwhelmingly large share of core members.
The distribution of onion roles in the communities of Elasticsearch and MongoDB both experience a shift at some point in time. In the project of MongoDB, the proportion of core members increases on expense of regular members. In the community of Elasticsearch, the share of casual members increases on expense of core members. The distribution of onion roles in the communities of Redis is stable throughout the history of the projects.
At the end of the available data, the shares in the communities of Elasticsearch and Redis are stable. In the project of MongoDB, the data suggests a new shift in shares in 2022.
Investigating the shares of the onion roles around the dates of license change leads to the following insights. In the community of MongoDB, the share of core members increased after the license change on expense of the share of regular members. However, we can observe an increasing trend even before the license change and the increase of the share of core members only equalizes a preceding local minimum. In the communities of Redis, there is a small increase of the share of core members between the two license changes. Similar to MongoDB, this increase equalizes a local minimum. Regarding Elasticsearch, there so indication of a change of the distribution of onion roles around the license change.
This analysis was conducted to examine proposition 4.

> *Proposition 4: Changing the license from an open source license to a cloud protection license leads to increased knowledge concentration considering individuals in the community.*

In the project of Elasticsearch the share of core members remained stable around the license change, as we can see in Figure 7.11. Regarding the community of MongoDB, we can observe an increased share in Figure 7.5. Similarly, the share of core members in the communities of Redis increased, as depicted in Figure 7.17.
Consequently, the analyzed data suggests no support for proposition 4.

Similar to the distribution of onion roles, the proportion of contributions authored by employees of the corresponding company is characteristic for each project.
In the project of MongoDB, we observed a very small share in the first months, followed by a sharp increase to more than 90%. Later, the share stabilized around 80%. Regarding Elasticsearch, the share is small in the beginning but steadily increased. Throughout the analyzed time period, the share remains smaller than in the project of MongoDB with values between 30% and 50% in most months. The data of Redis exposes even smaller shares, with large peaks in a few months, but even then the proportion is less than 40%. At the end of the analyzed time period, the share of contributions authored by employees of Redis fluctuates around 20%.
The varying shares reflect the role of the company in the project. In the project of MongoDB the involvement seems to be stronger than in the other two projects. We can infer reasons

for the varying degree of involvement from the history of the projects. MongoDB started the project out of a company, Elasticsearch formed company around the project, and the projects of Redis represent supporting services for their core product.
Considering the situation around the dates of license change, we determined that MongoDB changed its license after the corresponding share of contributions stabilized of a high level. In the following months we could observe a peak, but afterwards the share returned to the previous level. In the project of Elasticsearch, there is a decrease in the proportion in the two months directly following the license change, and subsequently an increase. Redis changed the license of its projects for the first time shortly after a peak of the share of contributions authored by employees. In the next months, we could observe smaller shares until the second license change. Then, the proportion increased.
We analyzed the share of contributions authored by employees of the respective project to examine proposition 5.

> *Proposition 5: Changing the license from an open source license to a cloud protection license leads to increased knowledge concentration considering organizations in the community.*

The analyzed data provides no contradicting evidence regarding proposition 5.
Considering the projects of Redis in particular, the share of contributions before the first license change the pattern shows fluctuations with large peaks. Therefore, the decrease from the peak occurring directly before the first license change must be interpreted as normal in the sense of the pattern. In contrast to previous months, the share does not decline to 0% after the first license change, which could be interpreted as increase in comparison to the development after previous peaks. After the second license change of Redis, there is a clearly increasing trend, as we can see in Figure 7.18.
Combined with the long-term increase observed in the community of Elasticsearch (see Figure 7.12), and the stable share in the project of MongoDB (see Figure 7.6), we can conclude that the data shows some support for proposition 5.

8 Conclusion and Discussion

The starting point for this thesis was the strategic adoption of cloud protection licenses by COSS companies. Aiming to prevent their competitors from creating copies of their SaaS products, these companies also abandon their open source roots with this decision. This raises questions about the potential impact on their respective communities, which evolved around open source principles.

Correspondingly, the main research question of this thesis was: *What is the impact on community health when a COSS company changes its licensing terms from an open source license to a cloud protection license?*

We decided to investigate the impact of the license change on community health by analyzing the development of the activity within and the structure of the communities. Based on literature, we presented our expectations and examined them using data extracted from the Git repositories and GitHub projects of MongoDB, Elasticsearch and Redis.

Contradicting our expectations, we found no support for an impact of the license change on community activity. Similarly, we could not find an indication of the proposed impact on the development of joining or leaving community members. Moreover, the data in the investigated projects contradicts the expectation of an increased knowledge concentration among community members after the license change. The only expectation which is supported by the data of the projects is the proposed increase of knowledge concentration considering organizations in the community.

To conclude, the data indicates that the impact of a license change to a cloud protection license on community health is rather small and constrained to the concentration of knowledge in respect to organizations.

8.1 Link between Thesis and Management of Technology

This thesis relates to the Management of Technology study program as follows. First, the thesis takes the perspective of the respective companies to analyze the effects of their strategic choices on their operations. Moreover, the investigated context is open source software, a field strongly following open innovation principles. Next, it examines a problem that is located at the intersection of technology, organizations, and strategy. As such it investigates the impact of a strategic choice of companies to use legal measures to protect a business which is necessary because of the applied technological principles (SaaS) on their open innovation approach.

8.2 Contributions

Regarding the practical problem introduced in chapter 1.2, executives of COSS companies can read this thesis to gain an understanding of the potential effects of the adoption of a cloud protection license on their community. According to the results presented in this thesis, there is no impact of such a license change on technical and social activity within the community. Similarly, they can expect that adopting a cloud protection license will not lead to less joining or more leaving community members. Furthermore, the data shows that there is no shift in the concentration of knowledge regarding individual community members. The only impact of the license change on the health of their respective community might be an increased knowledge concentration on their own company. As a result of this study, we can recommend executives to adopt a cloud protection license if it addresses the strategic needs of their companies. However, they must consider the specific context of their company which might imply additional concerns.

Furthermore, this thesis contributed to the research in the field of open source software. In chapter 3.1 and chapter 3.2, several papers were presented which focus on evaluating community activity or various aspects of community structure in OSS communities. This thesis applied similar approaches in the slightly different context of COSS.

It continues the research on the impact of choice and change of software licenses on open source communities with a focus on cloud protection licenses, which represent a new and unexplored category of licenses. This way, the thesis contributes to research on several aspects of open source community health.

As presented in chapter 3.3, Colazo and Fang (2009) and Stewart et al. (2006) analyzed the impact of software licenses on technical activity in OSS projects, without considering cloud protection licenses. The results of the analysis performed in this thesis indicate that a change to a cloud protection license does not impact technical activity in COSS projects. Similarly, by evaluating joining and leaving community members, the thesis contributes to the examination of the influence of software licenses on the attractiveness of open source projects, as performed by Santos (2017) for instance. Again, the main contribution is the focus on the category of cloud protection licenses, which was missing in previous research. The results presented in chapter 7.4 indicate that switching to a cloud protection license does not impact the attractiveness of the project for developers.

Furthermore, the thesis combines the research on open source community structures and license changes by including previously disregarded aspects of the structure of open source communities in the evaluation of the impact of the license change. In particular, it examines the development of the distribution of roles according to the widely used onion model, where no changes were induced by the license change. Additionally, it examines the development of the share of contributions authored by employees of the company which controls the project, where we could observe increasing shares after the license changes.

8.3 Limitations

The underlying theoretical concepts were selected from the literature in the area of OSS. Therefore, it could be that the transfer to COSS might not be perfect. Similarly, cloud protection licenses are a new phenomenon, and not open source licenses. Nevertheless, the formulated expectations were deducted from research on open source licenses since

this is the most closely related field of research.
Furthermore, the selected approach is a compromise of the research focus and the available data. With more time, it could be improved in various ways. For instance, additional data sources or more aspects of community health could be included.
The selection of the projects represent another limitation. First, we deliberately chose companies which operate in different contexts. Correspondingly, it is not surprising that the results diverge. Nevertheless, there are limitations related to the specific projects. Regarding Elasticsearch, the period after the license change is relatively short. This became clear especially when the analyzed data represents quarters. Here, only seven data points follow the event of the license change. Considering the projects of Redis, the absolute values are simply small. This makes it hard to draw meaningful conclusions.
As the results of this thesis depend on the functioning of CHAOSS, there are corresponding limitations. First, the creators of the tools state that the unification of identities might not work 100% correct. In our case, the individual members were mostly identified using their email addresses. Usually, CHAOSS detects when members change email addresses at some point in time and is able to associate the email addresses with the same individual, however, sometimes this could not work. We can assume that this only affects a small share of community members and therefore does not affect the overall outcome.
Similarly, when validating the results calculated by the Python scripts by comparing them with the numbers presented directly in CHAOSS, there were sometimes small differences, e.g., the values of joining members differed by 1. Again, these rare occurrences do not change any overall outcomes.
Last, especially the data considering social activity in the project of MongoDB is suspicious on first impression because of the small scale compared to its technical activity and the unusual development over time. These results were calculated using the exact same procedures as in the other two projects. Nevertheless, further investigations would be needed to ensure its validity.

8.4 Further Research

While this thesis builds on research on OSS to analyze COSS, little is known about the differences between these related areas for the involved individuals. For instance, the motivation for a person involved in COSS could be different from the motivation of a person in OSS. Therefore, future research could focus on strengthening the theoretical link between COSS and OSS, or determining the differences.
Continuing the applied research approach, more data sources could be included, for instance other communication platforms. Additional projects could be analyzed to test the results of this work. Furthermore, other aspects of community health, which we deliberately excluded, could be evaluated, e.g., the impact of a license change on the culture within the community.

References

Amrit, C., & Van Hillegersberg, J. (2010). Exploring the impact of socio-technical core-periphery structures in open source software development. *Journal of Information Technology*, *25*(2), 216–229. https://doi.org/10.1057/jit.2010.7

Armbrust, M., Fox, A., Griffith, R., Joseph, A., Katz, R., Konwinski, A., Lee, G., Patterson, D., Rabkin, A., Stoica, I., & Zaharia, M. (2010). A view of cloud computing. *Communications of the ACM*, *53*(4), 50–58. https://doi.org/10.1145/1721654.1721672

Arne-Kristian Groven, Kirsten Haaland, Ruediger Glott, Anna Tannenberg, & Xavier Darbousset-Chong. (2011, September 1). Quality assessment of FOSS. In *INF5780 compendium autumn 2011: Open source, open collaboration and innovation*.

Aué, J., Haisma, M., Tómasdóttir, K. F., & Bacchelli, A. (2016). Social diversity and growth levels of open source software projects on GitHub. *Proceedings of the 10th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement*, 1–6. https://doi.org/10.1145/2961111.2962633

AWS. (2015, October 1). *New – amazon elasticsearch service | AWS news blog* [Section: Amazon OpenSearch Service]. Retrieved February 2, 2023, from https://aws.amazon.com/blogs/aws/new-amazon-elasticsearch-service/

Blischak, J., Davenport, E., & Wilson, G. (2016). A quick introduction to version control with git and GitHub. *PLoS Computational Biology*, *12*(1). https://doi.org/10.1371/journal.pcbi.1004668

Cheng, J., & Guo, J. (2019). Activity-based analysis of open source software contributors: Roles and dynamics, 11–18. https://doi.org/10.1109/CHASE.2019.00011

Claes, M., Mäntylä, M., Kuutila, M., & Farooq, U. (2018). Towards automatically identifying paid open source developers. *Proceedings of the 15th International Conference on Mining Software Repositories*, 437–441. https://doi.org/10.1145/3196398.3196447

Cleveland, R. B., Cleveland, W. S., McRae, J. E., & Terpenning, I. (1990). STL: A seasonal-trend decomposition. *J. Off. Stat*, *6*(1), 3–73.

Colazo, J., & Fang, Y. (2009). Impact of license choice on open source software development activity. *Journal of the American Society for Information Science and Technology*, *60*(5), 997–1011. https://doi.org/10.1002/asi.21039

Cosentino, V., Izquierdo, J. L. C., & Cabot, J. (2015). Assessing the bus factor of git repositories [ISSN: 1534-5351]. *2015 IEEE 22nd International Conference on Software Analysis, Evolution, and Reengineering (SANER)*, 499–503. https://doi.org/10.1109/SANER.2015.7081864

Dhasade, A., Venigalla, A., & Chimalakonda, S. (2020). Towards prioritizing GitHub issues. https://doi.org/10.1145/3385032.3385052

Dias, L. F., Steinmacher, I., & Pinto, G. (2018). Who drives company-owned OSS projects: Internal or external members? [Number: 1 Publisher: SpringerOpen]. *Journal of the Brazilian Computer Society*, *24*(1), 1–17. https://doi.org/10.1186/s13173-018-0079-x

Dueñas, S., Cosentino, V., Gonzalez-Barahona, J. M., Felix, A. d. C. S., Izquierdo-Cortazar, D., Cañas-Díaz, L., & García-Plaza, A. P. (2021). GrimoireLab: A toolset for software development analytics [Publisher: PeerJ Inc.]. *PeerJ Computer Science*, *7*, e601. https://doi.org/10.7717/peerj-cs.601

Elasticsearch B.V. (2021). *FAQ on 2021 license change*. Retrieved October 27, 2022, from https://www.elastic.co/pricing/faq/licensing

FOSSA Inc. (2018, April 12). *Commons clause license* [Commons clause license]. Retrieved January 25, 2023, from https://commonsclause.com/

Free Software Foundation. (2007, June 29). *The GNU general public license v3.0 - GNU project - free software foundation*. Retrieved January 25, 2023, from https://www.gnu.org/licenses/gpl-3.0.html

Gamalielsson, J., & Lundell, B. (2014). Sustainability of open source software communities beyond a fork: How and why has the LibreOffice project evolved? *Journal of Systems and Software*, *89*, 128–145. https://doi.org/10.1016/j.jss.2013.11.1077

Github Inc. (2022). *GitHub - advanced search* [GitHub]. Retrieved February 2, 2023, from https://github.com/search/advanced

Goggins, S., Lumbard, K., & Germonprez, M. (2021). Open source community health: Analytical metrics and their corresponding narratives, 25–33. https://doi.org/10.1109/SoHeal52568.2021.00010

Hauge, Ø., & Ziemer, S. (2009). Providing commercial open source software: Lessons learned [ISBN: 9783642020315]. *IFIP Advances in Information and Communication Technology*, *299*, 70–82. https://doi.org/10.1007/978-3-642-02032-2_8

Hofmann, G., Riehle, D., Kolassa, C., & Mauerer, W. (2013). A dual model of open source license growth. In E. Petrinja, G. Succi, N. El Ioini, & A. Sillitti (Eds.), *Open source software: Quality verification* (pp. 245–256). Springer. https://doi.org/10.1007/978-3-642-38928-3_18

Igor Kotua. (2022, March 28). *Strong OSS licenses is the new black? understanding SSPL, ELv2, BSL and the commons clause* [Igor kotua]. Retrieved January 25, 2023, from https://igorkotua.com/strong-oss-licenses/

Linåker, J., Papatheocharous, E., & Olsson, T. (2022). How to characterize the health of an open source software project? a snowball literature review of an emerging practice. https://doi.org/10.1145/3555051.3555067

Linux Foundation. (2022a). *CHAOSS - metrics - release 2022-04* [CHAOSS - metrics - release 2022-04]. Retrieved December 5, 2022, from https://chaoss.community/wp-content/uploads/2022/04/English-Release-2022-04-18.pdf

Linux Foundation. (2022b). *Community health analytics in open source software* [CHAOSS]. Retrieved December 5, 2022, from https://chaoss.community/

MongoDB, Inc. (2018, October 16). *Server side public license (SSPL)* [MongoDB]. Retrieved December 1, 2022, from https://www.mongodb.com/licensing/server-side-public-license

MongoDB, Inc. (2022). *Server side public license FAQ* [MongoDB]. Retrieved October 27, 2022, from https://www.mongodb.com/licensing/server-side-public-license/faq

Nakakoji, K., Yamamoto, Y., Nishinaka, Y., Kishida, K., & Ye, Y. (2002). Evolution patterns of open-source software systems and communities. *Proceedings of the International Workshop on Principles of Software Evolution*, 76–85. https://doi.org/10.1145/512035.512055

Open Source Initiative. (2007). *The open source definition | open source initiative*. Retrieved October 27, 2022, from https://opensource.org/osd

Open Source Initiative. (2018). *History of the OSI | open source initiative*. Retrieved October 27, 2022, from https://opensource.org/history

Open Source Initiative. (2021, January 19). *The SSPL is not an open source license | open source initiative*. Retrieved December 1, 2022, from https://opensource.org/sspl-not-open-source

Open Source Initiative. (2022a). *Frequently answered questions | open source initiative*. Retrieved November 29, 2022, from https://opensource.org/faq#copyleft

Open Source Initiative. (2022b). *The MIT license | open source initiative*. Retrieved January 25, 2023, from https://opensource.org/licenses/MIT

Popp, K. (2012). Leveraging open source licenses and open source communities in hybrid commercial open source business models [ISSN: 1613-0073], *879*, 33–40.

Redis Ltd. (2022, November 15). *Redis source available license 2.0 (RSALv2) agreement* [Redis]. Retrieved December 1, 2022, from https://redis.com/legal/rsalv2-agreement/

Riehle, D. (2012). The single-vendor commercial open course business model. *Information Systems and e-Business Management*, *10*(1), 5–17. https://doi.org/10.1007/s10257-010-0149-x

Riehle, D. (2019). The innovations of open source [Conference Name: Computer]. *Computer*, *52*(4), 59–63. https://doi.org/10.1109/MC.2019.2898163

Riehle, D. (2020). Single-vendor open source firms [Conference Name: Computer]. *Computer*, *53*(4), 68–72. https://doi.org/10.1109/MC.2020.2969672

Riehle, D. (2021). The open source distributor business model [Conference Name: Computer]. *Computer*, *54*(12), 99–103. https://doi.org/10.1109/MC.2021.3112318

Riehle, D., Riemer, P., Kolassa, C., & Schmidt, M. (2014). Paid vs. volunteer work in open source [ISSN: 1530-1605]. *2014 47th Hawaii International Conference on System Sciences*, 3286–3295. https://doi.org/10.1109/HICSS.2014.407

Santos, C. D. d. (2017). Changes in free and open source software licenses: Managerial interventions and variations on project attractiveness. *Journal of Internet Services and Applications*, *8*(1), 11. https://doi.org/10.1186/s13174-017-0062-3

Shaikh, M., & Levina, N. (2019). Selecting an open innovation community as an alliance partner: Looking for healthy communities and ecosystems. *Research Policy*, *48*(8). https://doi.org/10.1016/j.respol.2019.03.011

Stewart, K. J., Ammeter, A. P., & Maruping, L. M. (2006). Impacts of license choice and organizational sponsorship on user interest and development activity in open source software projects [Publisher: INFORMS]. *Information Systems Research*, *17*(2), 126–144. https://doi.org/10.1287/isre.1060.0082

Timescale, Inc. (2020, September 24). *Building a self-sustaining open-source business in the cloud era* [Timescale blog]. Retrieved January 31, 2023, from https://www.timescale.com/blog/building-open-source-business-in-cloud-era-v2/

Torchiano, M., Ricca, F., & Marchetto, A. (2011). Is my project's truck factor low? theoretical and empirical considerations about the truck factor threshold. *Proceedings of the 2nd International Workshop on Emerging Trends in Software Metrics*, 12–18. https://doi.org/10.1145/1985374.1985379

Valiev, M., Vasilescu, B., & Herbsleb, J. (2018). Ecosystem-level determinants of sustained activity in open-source projects: A case study of the PyPI ecosystem. *Proceedings of the 2018 26th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, 644–655. https://doi.org/10.1145/3236024.3236062

Vendome, C., Bavota, G., Penta, M., Linares-Vásquez, M., German, D., & Poshyvanyk, D. (2017). License usage and changes: A large-scale study on gitHub. *Empirical Software Engineering*, *22*(3), 1537–1577. https://doi.org/10.1007/s10664-016-9438-4

Viseur, R., & Robles, G. (2015). First results about motivation and impact of license changes in open source projects [ISBN: 9783319178363]. *IFIP Advances in Information and Communication Technology*, *451*, 137–145. https://doi.org/10.1007/978-3-319-17837-0_13

Wang, Z., Feng, Y., Wang, Y., Jones, J., & Redmiles, D. (2020). Unveiling elite developers activities in open source projects. *ACM Transactions on Software Engineering and Methodology*, *29*(3). https://doi.org/10.1145/3387111

Weikert, F., & Riehle, D. (2013). A model of commercial open source software product features [ISBN: 9783642393358]. *Lecture Notes in Business Information Processing*, *150 LNBIP*, 90–101. https://doi.org/10.1007/978-3-642-39336-5_10

Wilson, P. (2013, January 1). Chapter 19 - intellectual property. In R. Oshana & M. Kraeling (Eds.), *Software engineering for embedded systems* (pp. 647–670). Newnes. https://doi.org/10.1016/B978-0-12-415917-4.00019-0

A Projects using a Cloud Protection License

Project	License
MongoDB Inc.	SSPL
Elastic NV	SSPL/ELv2
Graylog	SSPL
Netmaker	SSPL
Basin	SSPL
Tapdata	SSPL
whaleal	SSPL
ClearML	SSPL
Striveworks LIteCOW	SSPL
V12 Technology Fluxtion	SSPL
Datatorch	SSPL
deployed cc	SSPL
AriByte	ELv2
OpenReplay	ELv2
Invoiceninja	ELv2
Yatai	ELv2
Starrocks	ELv2
Apollo Graph	ELv2
Koi	Apache 2.0 with Commons Clause
Astronomer	Apache 2.0 with Commons Clause
vectorbt	Apache 2.0 with Commons Clause
n8n GmbH	Apache 2.0 with Commons Clause
Redis	Apache 2.0 with Commons Clause

Table A1: Projects on GitHub using a cloud protection license

B Data Tables

B.1 MongoDB

B.1.1 Community Activity

Table B1: Time series data of monthly community activity for MongoDB

date	Commits	Seasonal	Trend	Residual	Issues	Seasonal	Trend	Residual
2007-10-01	21	2.1	-92.2	111.1	0.0	0.0	0.0	0.0
2007-11-01	70	7.8	-70.3	132.5	0.0	0.0	0.0	0.0
2007-12-01	22	41.4	-47.7	28.3	0.0	0.0	0.0	0.0
2008-01-01	13	147.4	-24.7	-109.7	0.0	0.0	0.0	0.0
2008-02-01	52	224.0	-1.2	-170.8	0.0	0.0	0.0	0.0
2008-03-01	79	28.7	22.5	27.8	0.0	0.0	0.0	0.0
2008-04-01	19	16.4	46.5	-43.9	0.0	0.0	0.0	0.0
2008-05-01	6	-19.5	70.7	-45.2	0.0	0.0	0.0	0.0
2008-06-01	59	-101.3	95.2	65.1	0.0	0.0	0.0	0.0
2008-07-01	60	-46.8	120.3	-13.5	0.0	0.0	0.0	0.0
2008-08-01	45	-66.7	146.6	-34.9	0.0	0.0	0.0	0.0
2008-09-01	47	-120.1	176.7	-9.7	0.0	0.0	0.0	0.0
2008-10-01	42	-27.7	206.3	-136.6	0.0	0.0	0.0	0.0
2008-11-01	72	-42.0	231.5	-117.5	0.0	0.0	0.0	0.0
2008-12-01	324	41.8	252.6	29.6	0.0	0.0	0.0	0.0
2009-01-01	607	84.8	271.3	250.8	0.0	0.0	0.0	0.0
2009-02-01	721	160.7	289.5	270.8	0.0	0.0	0.0	0.0
2009-03-01	324	26.4	307.5	-9.9	0.0	0.0	0.0	0.0
2009-04-01	402	35.5	325.1	41.4	0.0	0.0	0.0	0.0
2009-05-01	455	14.6	342.2	98.2	0.0	0.0	0.0	0.0
2009-06-01	140	-81.7	358.3	-136.6	0.0	0.0	0.0	0.0
2009-07-01	169	15.4	372.9	-219.2	0.0	0.0	0.0	0.0
2009-08-01	296	-38.6	386.1	-51.5	0.0	0.0	0.0	0.0
2009-09-01	305	-97.5	399.7	2.8	0.0	0.0	0.0	0.0
2009-10-01	407	-52.3	417.7	41.6	0.0	0.0	0.0	0.0
2009-11-01	290	-84.1	442.6	-68.5	0.0	0.0	0.0	0.0
2009-12-01	476	50.9	473.6	-48.5	0.0	0.0	0.0	0.0
2010-01-01	427	31.9	507.7	-112.6	0.0	0.0	0.0	0.0
2010-02-01	693	104.4	541.4	47.2	0.0	0.0	0.0	0.0

Continued on next page

Table B1: Time series data of monthly community activity for MongoDB

date	Commits	Seasonal	Trend	Residual	Issues	Seasonal	Trend	Residual
2010-03-01	544	23.7	572.4	-52.1	0.0	0.0	0.0	0.0
2010-04-01	700	51.2	598.5	50.3	0.0	0.0	0.0	0.0
2010-05-01	597	48.3	618.1	-69.4	0.0	0.0	0.0	0.0
2010-06-01	649	-67.6	632.0	84.6	0.0	0.0	0.0	0.0
2010-07-01	1148	58.9	642.5	446.6	0.0	0.0	0.0	0.0
2010-08-01	833	-17.7	651.0	199.7	0.0	0.0	0.0	0.0
2010-09-01	601	-77.4	657.1	21.3	4.0	-0.6	1.6	3.0
2010-10-01	530	-57.3	659.6	-72.3	4.0	8.4	2.6	-7.0
2010-11-01	566	-108.1	657.7	16.4	3.0	-1.8	3.7	1.1
2010-12-01	632	63.9	651.7	-83.6	0.0	-6.7	4.7	2.0
2011-01-01	532	-38.5	642.3	-71.9	10.0	0.2	5.7	4.0
2011-02-01	417	27.5	630.3	-240.8	4.0	-0.4	6.8	-2.4
2011-03-01	719	15.1	617.8	86.1	10.0	0.9	7.8	1.3
2011-04-01	664	54.2	607.7	2.1	2.0	-3.2	8.8	-3.6
2011-05-01	745	63.1	602.7	79.2	3.0	-6.0	9.8	-0.9
2011-06-01	593	-53.6	603.2	43.4	16.0	5.3	10.8	-0.1
2011-07-01	624	107.1	607.0	-90.1	17.0	4.1	11.8	1.2
2011-08-01	578	-0.9	612.0	-33.1	5.0	-2.3	12.7	-5.4
2011-09-01	584	-68.1	616.9	35.2	15.0	-0.2	13.5	1.8
2011-10-01	586	1.2	620.5	-35.7	43.0	9.5	14.3	19.2
2011-11-01	486	-50.5	621.7	-85.1	13.0	-1.7	15.1	-0.5
2011-12-01	778	92.6	620.1	65.3	7.0	-3.0	15.9	-5.9
2012-01-01	562	-78.8	616.7	24.1	12.0	0.6	16.4	-5.0
2012-02-01	700	-63.5	613.5	149.9	16.0	-0.3	16.6	-0.3
2012-03-01	608	26.3	611.3	-29.6	15.0	-1.7	16.7	0.0
2012-04-01	666	20.0	609.7	36.3	19.0	-4.4	16.6	6.9
2012-05-01	684	50.1	607.7	26.1	9.0	-5.4	16.4	-2.0
2012-06-01	556	-73.1	604.1	25.0	28.0	2.4	16.3	9.3
2012-07-01	578	40.9	598.4	-61.3	19.0	2.1	16.3	0.5
2012-08-01	464	-21.6	590.9	-105.3	23.0	-0.5	16.5	7.0
2012-09-01	506	-64.1	582.0	-12.0	9.0	0.6	16.7	-8.3
2012-10-01	644	112.9	573.6	-42.5	14.0	11.4	16.8	-14.2
2012-11-01	641	6.8	567.4	66.8	12.0	-1.7	16.8	-3.1
2012-12-01	774	97.8	564.3	111.9	19.0	0.0	16.8	2.2
2013-01-01	510	-69.7	564.7	15.1	17.0	0.9	16.8	-0.7

Continued on next page

Table B1: Time series data of monthly community activity for MongoDB

date	Commits	Seasonal	Trend	Residual	Issues	Seasonal	Trend	Residual
2013-02-01	422	-88.5	568.0	-57.5	22.0	-0.7	16.9	5.8
2013-03-01	514	45.5	573.2	-104.8	14.0	-3.9	17.2	0.7
2013-04-01	500	11.4	579.3	-90.7	14.0	-5.3	17.8	1.5
2013-05-01	580	23.1	585.4	-28.5	21.0	-5.1	18.6	7.5
2013-06-01	412	-116.4	592.0	-63.6	7.0	0.4	19.4	-12.8
2013-07-01	611	-33.0	599.6	44.5	20.0	0.6	20.1	-0.7
2013-08-01	673	-24.9	609.0	89.0	20.0	1.1	20.6	-1.7
2013-09-01	490	-39.4	620.7	-91.3	21.0	1.5	20.9	-1.4
2013-10-01	924	187.8	634.2	102.0	29.0	11.7	21.0	-3.7
2013-11-01	742	42.7	648.3	51.0	26.0	-1.7	21.3	6.5
2013-12-01	650	66.1	662.4	-78.5	35.0	3.1	21.6	10.3
2014-01-01	549	-62.4	677.0	-65.6	27.0	1.8	22.3	3.0
2014-02-01	574	-97.7	692.2	-20.4	25.0	-1.1	23.0	3.1
2014-03-01	894	62.6	708.3	123.1	12.0	-5.6	23.7	-6.2
2014-04-01	712	35.9	725.5	-49.5	8.0	-6.2	24.3	-10.1
2014-05-01	686	-20.7	743.2	-36.5	18.0	-4.3	24.6	-2.4
2014-06-01	629	-145.9	760.6	14.3	20.0	-1.8	24.7	-2.9
2014-07-01	658	-36.0	777.6	-83.5	20.0	-0.6	24.5	-3.9
2014-08-01	676	35.4	794.1	-153.5	35.0	1.9	24.2	8.9
2014-09-01	856	-4.9	809.7	51.1	37.0	3.2	23.9	9.9
2014-10-01	1220	151.2	824.1	244.7	51.0	10.5	23.6	16.9
2014-11-01	887	29.6	837.5	19.9	19.0	-1.4	23.2	-2.8
2014-12-01	866	7.6	849.9	8.5	24.0	2.8	22.7	-1.5
2015-01-01	892	-71.5	860.5	103.0	25.0	2.7	21.9	0.4
2015-02-01	690	-81.4	868.0	-96.6	11.0	-2.8	20.8	-7.0
2015-03-01	920	91.4	871.1	-42.6	16.0	-4.8	19.4	1.4
2015-04-01	1038	78.7	869.7	89.6	18.0	-5.9	17.9	6.0
2015-05-01	914	-62.4	864.6	111.7	7.0	-3.6	16.3	-5.7
2015-06-01	677	-112.9	856.7	-66.8	21.0	0.0	14.8	6.1
2015-07-01	826	-19.6	846.6	-1.0	15.0	-0.0	13.5	1.5
2015-08-01	1069	71.9	834.8	162.3	6.0	0.7	12.4	-7.1
2015-09-01	922	20.5	820.7	80.8	14.0	3.2	11.5	-0.7
2015-10-01	704	62.3	803.5	-161.9	16.0	8.4	10.7	-3.1
2015-11-01	781	-11.1	783.2	8.8	7.0	-2.1	9.9	-0.7
2015-12-01	743	-41.9	760.0	24.9	4.0	1.1	9.2	-6.3

Continued on next page

Table B1: Time series data of monthly community activity for MongoDB

date	Commits	Seasonal	Trend	Residual	Issues	Seasonal	Trend	Residual
2016-01-01	617	-66.2	733.9	-50.7	11.0	2.6	8.5	-0.1
2016-02-01	838	-63.0	705.9	195.1	6.0	-3.5	7.9	1.6
2016-03-01	827	85.1	677.3	64.6	6.0	-2.3	7.4	0.9
2016-04-01	790	108.9	648.7	32.4	1.0	-2.8	7.0	-3.2
2016-05-01	418	-56.2	619.9	-145.7	10.0	-2.4	6.7	5.7
2016-06-01	512	-35.7	591.5	-43.8	8.0	2.1	6.5	-0.6
2016-07-01	625	4.8	564.1	56.1	9.0	1.1	6.2	1.7
2016-08-01	630	101.6	537.8	-9.4	7.0	-1.4	6.1	2.3
2016-09-01	522	13.5	513.5	-5.0	3.0	0.6	5.9	-3.5
2016-10-01	350	-32.9	492.6	-109.7	2.0	3.3	5.7	-7.1
2016-11-01	363	-38.1	475.7	-74.6	5.0	-2.6	5.6	2.0
2016-12-01	399	-77.7	463.2	13.4	8.0	0.4	5.5	2.1
2017-01-01	372	-63.8	455.4	-19.6	6.0	2.7	5.4	-2.1
2017-02-01	296	-35.1	451.3	-120.3	1.0	-1.3	5.4	-3.1
2017-03-01	458	69.9	449.7	-61.7	5.0	-0.3	5.4	-0.1
2017-04-01	538	94.5	450.1	-6.6	8.0	-0.9	5.6	3.3
2017-05-01	388	-32.6	453.1	-32.5	3.0	0.1	5.8	-3.0
2017-06-01	596	51.1	458.9	86.0	10.0	0.7	6.1	3.1
2017-07-01	496	21.8	466.9	7.3	7.0	1.5	6.4	-0.9
2017-08-01	552	91.3	476.7	-16.1	3.0	-1.9	6.7	-1.8
2017-09-01	417	-22.0	487.7	-48.7	6.0	-1.6	6.9	0.6
2017-10-01	493	-61.0	499.1	54.9	8.0	0.2	7.1	0.7
2017-11-01	498	-63.2	510.3	50.9	2.0	-2.0	7.2	-3.1
2017-12-01	309	-109.3	520.4	-102.1	10.0	1.2	7.1	1.7
2018-01-01	485	-19.7	528.9	-24.2	12.0	2.3	7.0	2.7
2018-02-01	519	-36.7	536.1	19.6	12.0	0.5	6.9	4.6
2018-03-01	649	62.6	542.7	43.7	9.0	-0.2	6.7	2.5
2018-04-01	604	79.4	548.7	-24.0	5.0	0.0	6.5	-1.5
2018-05-01	681	10.2	553.7	117.1	10.0	1.1	6.3	2.7
2018-06-01	678	103.1	556.8	18.1	1.0	-0.8	6.0	-4.2
2018-07-01	533	5.1	557.4	-29.5	7.0	0.5	5.6	0.9
2018-08-01	632	57.4	555.5	19.0	3.0	-1.7	5.2	-0.5
2018-09-01	534	-53.6	552.4	35.2	1.0	-1.3	4.9	-2.5
2018-10-01	495	-44.1	548.8	-9.7	4.0	0.4	4.5	-0.9
2018-11-01	462	-84.2	545.3	0.9	4.0	-1.9	4.2	1.6

Continued on next page

Table B1: Time series data of monthly community activity for MongoDB

date	Commits	Seasonal	Trend	Residual	Issues	Seasonal	Trend	Residual
2018-12-01	503	-164.4	542.7	124.7	4.0	1.4	4.1	-1.5
2019-01-01	503	27.8	541.6	-66.4	6.0	1.8	4.0	0.2
2019-02-01	497	-30.0	542.1	-15.1	4.0	1.7	4.0	-1.7
2019-03-01	489	78.2	544.5	-133.7	2.0	-0.3	4.1	-1.7
2019-04-01	542	80.1	550.1	-88.2	6.0	-0.2	4.2	2.0
2019-05-01	531	81.9	559.2	-110.0	4.0	1.4	4.2	-1.6
2019-06-01	729	117.8	572.1	39.1	2.0	-1.7	4.3	-0.6
2019-07-01	614	-33.1	589.2	57.9	4.0	-0.5	4.3	0.2
2019-08-01	600	31.9	609.2	-41.1	4.0	-1.4	4.4	1.0
2019-09-01	494	-62.1	630.0	-73.9	5.0	-0.2	4.4	0.8
2019-10-01	614	-16.5	649.7	-19.2	8.0	0.2	4.5	3.2
2019-11-01	526	-101.6	667.4	-39.8	3.0	-0.9	4.6	-0.6
2019-12-01	428	-190.0	682.4	-64.4	6.0	0.7	4.6	0.7
2020-01-01	999	6.1	694.1	298.8	3.0	1.4	4.5	-3.0
2020-02-01	711	-21.6	702.4	30.2	7.0	2.1	4.4	0.5
2020-03-01	945	80.7	707.3	157.0	3.0	-0.1	4.3	-1.2
2020-04-01	944	80.7	709.0	154.2	3.0	-1.1	4.2	-0.1
2020-05-01	834	113.4	708.7	11.9	8.0	-0.1	4.0	4.1
2020-06-01	665	127.1	706.5	-168.6	4.0	-1.2	3.9	1.4
2020-07-01	577	-66.2	702.2	-59.0	1.0	-1.0	3.8	-1.8
2020-08-01	747	14.6	697.7	34.7	2.0	-0.6	3.7	-1.1
2020-09-01	694	-54.0	695.5	52.5	5.0	1.1	3.6	0.2
2020-10-01	625	33.6	697.3	-105.9	1.0	-0.2	3.6	-2.4
2020-11-01	576	-72.8	703.2	-54.3	3.0	-0.3	3.5	-0.2
2020-12-01	414	-158.6	713.0	-140.4	4.0	0.8	3.4	-0.2
2021-01-01	506	1.4	726.1	-221.5	6.0	0.8	3.4	1.8
2021-02-01	729	-139.6	741.9	126.7	5.0	2.4	3.4	-0.9
2021-03-01	794	90.3	759.1	-55.4	4.0	0.2	3.6	0.3
2021-04-01	754	95.1	776.3	-117.4	1.0	-2.0	3.7	-0.7
2021-05-01	1098	132.7	793.6	171.7	1.0	-1.6	3.9	-1.3
2021-06-01	1186	117.0	811.7	257.3	2.0	-0.5	4.1	-1.5
2021-07-01	771	-101.3	829.8	42.6	3.0	-1.4	4.3	0.2
2021-08-01	783	8.7	845.4	-71.1	5.0	0.1	4.5	0.5
2021-09-01	739	-32.3	856.0	-84.6	8.0	2.4	4.7	1.0
2021-10-01	1122	74.6	860.7	186.7	4.0	-0.6	4.9	-0.2

Continued on next page

Table B1: Time series data of monthly community activity for MongoDB

date	Commits	Seasonal	Trend	Residual	Issues	Seasonal	Trend	Residual
2021-11-01	800	-32.1	860.1	-27.9	7.0	0.2	5.1	1.7
2021-12-01	769	-121.6	855.5	35.1	4.0	1.2	5.3	-2.5
2022-01-01	749	1.1	848.1	-100.2	7.0	0.2	5.5	1.3
2022-02-01	777	-284.8	838.9	222.9	9.0	2.8	5.7	0.6
2022-03-01	931	98.7	829.5	2.9	7.0	0.6	5.8	0.6
2022-04-01	972	107.3	820.3	44.5	3.0	-3.1	5.9	0.2
2022-05-01	859	154.8	808.8	-104.6	2.0	-3.3	6.0	-0.7
2022-06-01	799	112.9	796.0	-109.9	7.0	0.2	6.1	0.7
2022-07-01	644	-136.0	782.4	-2.4	5.0	-1.8	6.2	0.6
2022-08-01	829	3.6	768.5	56.9	7.0	0.8	6.3	-0.1
2022-09-01	810	-9.2	754.4	64.8	9.0	3.6	6.4	-1.0
2022-10-01	789	121.7	740.4	-73.1	6.0	-1.1	6.5	0.7
2022-11-01	809	15.0	726.3	67.7	6.0	0.6	6.6	-1.2
2022-12-01	700	-74.0	712.3	61.7	10.0	1.5	6.6	1.8
2023-01-01	836	-0.5	698.3	138.2	5.0	-0.4	6.7	-1.3
2023-02-01	27	-442.4	684.2	-214.9	0.0	0.0	0.0	0.0

B.1.2 Community Structure

Table B2: Time series data of monthly joining and leaving members for MongoDB

date	Joining	Seasonal	Trend	Residual	Leaving	Seasonal	Trend	Residual
2007-10-01	2.00	-0.39	1.43	0.96	0.00	0.00	0.00	0.00
2007-11-01	1.00	0.57	1.44	-1.01	0.00	0.00	0.00	0.00
2008-02-01	1.00	-0.62	1.45	0.17	0.00	0.00	0.00	0.00
2008-03-01	2.00	-0.04	1.46	0.57	1.00	0.24	0.46	0.30
2008-07-01	1.00	-0.20	1.48	-0.28	0.00	0.00	0.00	0.00
2008-08-01	2.00	0.05	1.50	0.44	1.00	0.36	0.62	0.02
2008-10-01	1.00	-0.20	1.52	-0.32	0.00	0.00	0.00	0.00
2008-11-01	1.00	-0.18	1.55	-0.37	0.00	0.00	0.00	0.00
2008-12-01	1.00	-0.91	1.58	0.33	0.00	0.00	0.00	0.00
2009-01-01	0.00	0.00	0.00	0.00	1.00	2.18	0.78	-1.96
2009-02-01	2.00	1.16	1.60	-0.77	2.00	0.98	0.93	0.09

Continued on next page

Table B2: Time series data of monthly joining and leaving members for MongoDB

date	Joining	Seasonal	Trend	Residual	Leaving	Seasonal	Trend	Residual
2009-04-01	1.00	0.42	1.63	-1.05	1.00	-0.24	1.08	0.16
2009-05-01	2.00	0.15	1.66	0.19	1.00	-1.03	1.23	0.80
2009-06-01	1.00	0.02	1.73	-0.75	1.00	0.04	1.37	-0.41
2009-08-01	4.00	0.35	1.82	1.82	3.00	0.63	1.52	0.86
2009-09-01	1.00	-0.41	1.94	-0.53	0.00	0.00	0.00	0.00
2009-10-01	2.00	-0.01	2.06	-0.05	2.00	-0.96	1.66	1.29
2009-11-01	0.00	0.00	0.00	0.00	1.00	-1.76	1.81	0.95
2009-12-01	2.00	-0.14	2.17	-0.04	1.00	0.26	1.96	-1.23
2010-01-01	1.00	-0.31	2.26	-0.96	0.00	0.00	0.00	0.00
2010-02-01	2.00	-0.53	2.33	0.20	2.00	0.14	2.11	-0.25
2010-03-01	3.00	-0.02	2.37	0.65	2.00	-0.37	2.17	0.20
2010-04-01	1.00	-1.08	2.39	-0.32	1.00	0.81	2.20	-2.01
2010-05-01	5.00	1.65	2.41	0.94	6.00	1.63	2.21	2.16
2010-06-01	5.00	0.36	2.45	2.19	4.00	0.59	2.24	1.17
2010-07-01	3.00	-0.08	2.50	0.58	2.00	-0.15	2.30	-0.15
2010-08-01	1.00	0.42	2.57	-1.99	1.00	-0.93	2.39	-0.47
2010-09-01	2.00	0.20	2.62	-0.81	3.00	-0.49	2.52	0.97
2010-10-01	2.00	-0.27	2.65	-0.38	1.00	0.02	2.65	-1.67
2010-11-01	2.00	0.11	2.67	-0.78	1.00	-0.35	2.75	-1.40
2010-12-01	2.00	-0.19	2.71	-0.52	0.00	0.00	0.00	0.00
2011-01-01	2.00	-0.73	2.77	-0.04	1.00	-0.94	2.81	-0.87
2011-02-01	3.00	-0.82	2.88	0.95	0.00	0.00	0.00	0.00
2011-03-01	4.00	0.24	3.03	0.73	4.00	0.80	2.85	0.35
2011-05-01	2.00	-1.17	3.24	-0.07	3.00	0.20	2.89	-0.09
2011-06-01	5.00	2.07	3.46	-0.53	2.00	-0.86	2.95	-0.09
2011-08-01	2.00	0.31	3.66	-1.98	0.00	0.00	0.00	0.00
2011-09-01	2.00	-0.25	3.83	-1.58	0.00	0.00	0.00	0.00
2011-10-01	7.00	0.80	3.98	2.22	8.00	1.10	3.04	3.85
2011-11-01	4.00	-0.05	4.13	-0.08	6.00	1.06	3.18	1.76
2011-12-01	6.00	-0.14	4.29	1.85	2.00	0.34	3.33	-1.67
2012-01-01	3.00	0.30	4.47	-1.77	2.00	-0.16	3.46	-1.30
2012-02-01	7.00	-0.36	4.64	2.72	1.00	-0.83	3.53	-1.70
2012-03-01	5.00	-1.11	4.81	1.30	2.00	-0.99	3.55	-0.57
2012-04-01	2.00	-1.07	4.97	-1.89	4.00	-0.62	3.55	1.08
2012-05-01	3.00	0.61	5.08	-2.69	3.00	0.26	3.55	-0.81

Continued on next page

Table B2: Time series data of monthly joining and leaving members for MongoDB

date	Joining	Seasonal	Trend	Residual	Leaving	Seasonal	Trend	Residual
2012-06-01	4.00	-1.07	5.16	-0.09	4.00	-0.01	3.58	0.44
2012-07-01	9.00	2.35	5.20	1.45	8.00	1.26	3.63	3.10
2012-08-01	7.00	0.07	5.23	1.70	5.00	0.22	3.72	1.06
2012-09-01	5.00	-0.45	5.25	0.21	1.00	-1.23	3.80	-1.57
2012-10-01	7.00	0.69	5.25	1.06	4.00	1.49	3.83	-1.32
2012-11-01	4.00	0.37	5.24	-1.61	2.00	0.34	3.80	-2.13
2012-12-01	4.00	-0.14	5.22	-1.08	3.00	0.05	3.70	-0.76
2013-01-01	9.00	0.27	5.20	3.53	7.00	-0.28	3.59	3.69
2013-02-01	3.00	-0.99	5.18	-1.20	5.00	-0.75	3.49	2.26
2013-03-01	3.00	-1.24	5.16	-0.92	2.00	-1.56	3.43	0.13
2013-04-01	5.00	-1.52	5.14	1.38	2.00	-1.14	3.39	-0.25
2013-05-01	7.00	1.71	5.13	0.16	5.00	0.91	3.37	0.72
2013-06-01	2.00	-0.04	5.13	-3.09	2.00	1.12	3.37	-2.49
2013-07-01	7.00	2.03	5.12	-0.15	3.00	1.72	3.36	-2.08
2013-08-01	4.00	-0.32	5.13	-0.81	3.00	0.25	3.35	-0.60
2013-09-01	6.00	-0.08	5.16	0.92	2.00	-0.49	3.35	-0.86
2013-10-01	5.00	-0.18	5.23	-0.05	5.00	0.33	3.40	1.27
2013-11-01	8.00	0.58	5.34	2.07	3.00	-0.79	3.50	0.29
2013-12-01	5.00	-0.30	5.47	-0.18	5.00	0.21	3.67	1.11
2014-01-01	5.00	0.07	5.58	-0.66	1.00	-0.55	3.89	-2.33
2014-02-01	3.00	-1.64	5.66	-1.03	3.00	-0.42	4.11	-0.69
2014-03-01	3.00	-1.26	5.71	-1.45	0.00	0.00	0.00	0.00
2014-04-01	0.00	0.00	0.00	0.00	2.00	-1.95	4.33	-0.38
2014-05-01	4.00	-1.87	5.71	0.16	3.00	-1.44	4.53	-0.09
2014-06-01	10.00	3.73	5.68	0.60	8.00	1.37	4.70	1.94
2014-07-01	11.00	0.91	5.63	4.46	9.00	3.11	4.83	1.06
2014-08-01	7.00	1.42	5.60	-0.02	6.00	1.28	4.94	-0.22
2014-09-01	6.00	-0.95	5.57	1.39	5.00	0.11	5.01	-0.12
2014-10-01	4.00	0.49	5.51	-2.00	7.00	0.52	5.05	1.43
2014-11-01	2.00	-1.11	5.42	-2.31	4.00	-1.14	5.07	0.07
2014-12-01	7.00	0.23	5.27	1.50	4.00	-1.26	5.07	0.19
2015-01-01	5.00	-0.25	5.10	0.15	6.00	0.06	5.05	0.89
2015-02-01	3.00	-0.88	4.93	-1.05	4.00	-1.22	5.01	0.21
2015-03-01	3.00	-1.78	4.78	-0.00	4.00	-0.31	4.96	-0.65
2015-04-01	6.00	-1.42	4.68	2.74	3.00	-2.14	4.91	0.23

Continued on next page

Table B2: Time series data of monthly joining and leaving members for MongoDB

date	Joining	Seasonal	Trend	Residual	Leaving	Seasonal	Trend	Residual
2015-05-01	2.00	-2.36	4.63	-0.27	2.00	-1.41	4.85	-1.44
2015-06-01	10.00	6.68	4.63	-1.30	4.00	1.25	4.79	-2.04
2015-07-01	4.00	1.80	4.64	-2.44	10.00	5.67	4.72	-0.40
2015-08-01	5.00	0.56	4.63	-0.19	8.00	0.71	4.68	2.61
2015-09-01	1.00	-0.68	4.59	-2.91	0.00	0.00	0.00	0.00
2015-10-01	8.00	0.47	4.57	2.96	5.00	0.01	4.68	0.31
2015-11-01	5.00	-2.11	4.57	2.53	7.00	0.90	4.72	1.38
2015-12-01	2.00	-0.37	4.60	-2.24	1.00	-1.65	4.76	-2.11
2016-01-01	5.00	-0.45	4.64	0.81	3.00	-1.52	4.80	-0.27
2016-02-01	4.00	-1.62	4.67	0.94	4.00	-0.58	4.80	-0.22
2016-03-01	4.00	-1.50	4.69	0.80	3.00	-1.36	4.76	-0.40
2016-04-01	2.00	-1.40	4.68	-1.28	5.00	-0.42	4.68	0.75
2016-05-01	1.00	-2.61	4.63	-1.02	2.00	-1.94	4.56	-0.61
2016-06-01	13.00	9.46	4.54	-1.00	5.00	-1.52	4.42	2.10
2016-07-01	9.00	1.10	4.45	3.45	7.00	0.58	4.30	2.12
2016-08-01	5.00	-0.49	4.38	1.11	13.00	7.43	4.21	1.36
2016-09-01	4.00	0.19	4.37	-0.56	2.00	0.56	4.13	-2.69
2016-10-01	3.00	0.73	4.40	-2.13	4.00	0.03	4.03	-0.06
2016-11-01	1.00	-2.65	4.45	-0.80	4.00	0.30	3.92	-0.22
2016-12-01	3.00	-0.97	4.53	-0.56	1.00	-0.96	3.82	-1.86
2017-01-01	4.00	-1.14	4.64	0.50	2.00	-1.93	3.74	0.19
2017-02-01	1.00	-1.23	4.76	-2.53	1.00	-1.03	3.69	-1.66
2017-03-01	3.00	-1.47	4.88	-0.41	3.00	-1.15	3.68	0.47
2017-04-01	3.00	-2.03	5.00	0.03	4.00	-0.79	3.72	1.07
2017-05-01	3.00	-1.39	5.14	-0.75	2.00	-1.20	3.80	-0.60
2017-06-01	22.00	10.86	5.29	5.85	2.00	-1.48	3.90	-0.43
2017-07-01	3.00	0.27	5.44	-2.71	2.00	0.92	4.01	-2.93
2017-08-01	3.00	-1.04	5.55	-1.51	14.00	7.14	4.09	2.77
2017-09-01	10.00	0.56	5.59	3.84	5.00	-0.30	4.15	1.15
2017-10-01	7.00	0.42	5.59	0.99	4.00	-0.27	4.19	0.08
2017-11-01	1.00	-3.01	5.54	-1.53	3.00	-0.47	4.22	-0.75
2017-12-01	6.00	-1.14	5.44	1.69	7.00	0.75	4.24	2.01
2018-01-01	2.00	-1.96	5.32	-1.36	2.00	-2.01	4.27	-0.26
2018-02-01	6.00	-0.30	5.18	1.12	4.00	-0.87	4.28	0.59
2018-03-01	4.00	-1.86	5.04	0.81	3.00	-0.95	4.28	-0.33

Continued on next page

Table B2: Time series data of monthly joining and leaving members for MongoDB

date	Joining	Seasonal	Trend	Residual	Leaving	Seasonal	Trend	Residual
2018-04-01	3.00	-2.63	4.93	0.70	2.00	-1.60	4.24	-0.64
2018-05-01	3.00	0.77	4.82	-2.59	4.00	-0.08	4.17	-0.10
2018-06-01	14.00	11.51	4.71	-2.22	1.00	-1.18	4.08	-1.90
2018-07-01	5.00	-0.98	4.61	1.36	6.00	1.25	4.00	0.75
2018-08-01	2.00	-1.27	4.57	-1.30	10.00	5.21	3.93	0.85
2018-09-01	5.00	0.25	4.58	0.17	0.00	0.00	0.00	0.00
2018-10-01	5.00	0.10	4.63	0.27	4.00	-0.82	3.89	0.93
2018-11-01	2.00	-2.80	4.73	0.07	4.00	-0.94	3.88	1.06
2018-12-01	3.00	-1.35	4.85	-0.49	2.00	-0.67	3.89	-1.22
2019-01-01	3.00	-2.51	4.97	0.54	6.00	2.31	3.89	-0.20
2019-02-01	5.00	1.37	5.09	-1.46	1.00	-1.41	3.88	-1.47
2019-03-01	2.00	-2.03	5.18	-1.16	3.00	-0.53	3.85	-0.32
2019-04-01	1.00	-2.61	5.25	-1.64	3.00	-0.92	3.85	0.07
2019-05-01	14.00	1.68	5.32	6.99	1.00	-2.34	3.89	-0.55
2019-06-01	15.00	11.07	5.42	-1.49	6.00	0.46	3.99	1.55
2019-07-01	3.00	-1.70	5.54	-0.84	4.00	0.68	4.14	-0.82
2019-08-01	7.00	-1.59	5.66	2.93	8.00	0.46	4.34	3.21
2019-09-01	2.00	-0.46	5.78	-3.32	0.00	0.00	0.00	0.00
2019-10-01	5.00	0.14	5.87	-1.01	4.00	3.45	4.54	-3.99
2019-11-01	4.00	-1.88	5.92	-0.05	1.00	-0.80	4.72	-2.92
2019-12-01	4.00	-1.74	5.95	-0.21	2.00	-1.19	4.86	-1.67
2020-01-01	2.00	-2.10	5.97	-1.86	6.00	-0.63	4.97	1.66
2020-02-01	11.00	1.23	6.00	3.76	10.00	1.53	5.07	3.40
2020-03-01	4.00	-2.30	6.09	0.22	5.00	-0.32	5.17	0.14
2020-04-01	4.00	-3.04	6.22	0.82	7.00	-1.35	5.28	3.07
2020-05-01	6.00	0.90	6.39	-1.29	5.00	-0.94	5.37	0.57
2020-06-01	19.00	12.61	6.57	-0.18	2.00	-1.44	5.45	-2.02
2020-07-01	4.00	-1.69	6.77	-1.07	7.00	0.92	5.49	0.59
2020-08-01	4.00	-1.11	6.94	-1.83	7.00	2.88	5.47	-1.35
2020-09-01	7.00	-0.27	7.07	0.19	5.00	-0.43	5.37	0.05
2020-10-01	7.00	-0.10	7.17	-0.07	7.00	1.72	5.23	0.05
2020-11-01	7.00	-1.99	7.22	1.78	6.00	-1.00	5.09	1.91
2020-12-01	4.00	-1.96	7.25	-1.29	3.00	-1.40	4.97	-0.57
2021-01-01	7.00	-1.82	7.29	1.53	4.00	-0.54	4.90	-0.36
2021-02-01	10.00	0.74	7.35	1.91	5.00	0.57	4.87	-0.44

Continued on next page

Table B2: Time series data of monthly joining and leaving members for MongoDB

date	Joining	Seasonal	Trend	Residual	Leaving	Seasonal	Trend	Residual
2021-03-01	6.00	-2.71	7.44	1.27	7.00	0.70	4.88	1.42
2021-04-01	6.00	-3.66	7.53	2.14	1.00	-2.16	4.90	-1.74
2021-05-01	7.00	-0.21	7.59	-0.38	3.00	-0.90	4.92	-1.01
2021-06-01	18.00	14.83	7.61	-4.44	4.00	-0.34	4.94	-0.60
2021-07-01	6.00	-1.66	7.58	0.09	5.00	1.25	4.97	-1.22
2021-08-01	4.00	-0.39	7.51	-3.12	12.00	5.26	5.07	1.68
2021-09-01	8.00	0.13	7.46	0.41	2.00	-1.53	5.20	-1.67
2021-10-01	9.00	-0.45	7.44	2.02	7.00	0.15	5.34	1.50
2021-11-01	6.00	-2.26	7.48	0.79	4.00	-1.11	5.50	-0.39
2021-12-01	8.00	-2.35	7.58	2.76	5.00	-1.57	5.66	0.91
2022-01-01	6.00	-1.43	7.74	-0.30	5.00	-0.48	5.83	-0.34
2022-02-01	5.00	0.04	7.84	-2.88	4.00	-0.64	5.99	-1.35
2022-03-01	4.00	-3.08	7.88	-0.80	7.00	1.79	6.16	-0.94
2022-04-01	2.00	-4.27	7.90	-1.63	3.00	-3.15	6.33	-0.18
2022-05-01	6.00	-1.61	7.91	-0.30	6.00	-0.89	6.50	0.39
2022-06-01	29.00	17.25	7.91	3.83	9.00	0.93	6.67	1.40
2022-07-01	7.00	-1.54	7.93	0.61	0.00	0.00	0.00	0.00
2022-08-01	11.00	0.33	7.94	2.74	0.00	0.00	0.00	0.00
2022-09-01	9.00	0.73	7.95	0.32	0.00	0.00	0.00	0.00
2022-10-01	6.00	-0.76	7.96	-1.20	0.00	0.00	0.00	0.00
2022-11-01	4.00	-2.65	7.96	-1.31	0.00	0.00	0.00	0.00
2022-12-01	4.00	-2.66	7.96	-1.30	0.00	0.00	0.00	0.00

Table B3: Time series data of quarterly onion roles for MongoDB

quarter	Total Members	Core	Share	Regular	Share	Casual	Share
2010Q3	8	6	0.75	0	0.00	2	0.25
2010Q4	10	6	0.60	2	0.20	2	0.20
2011Q1	24	14	0.58	6	0.25	4	0.17
2011Q2	20	10	0.50	6	0.30	4	0.20
2011Q3	38	22	0.58	12	0.32	4	0.11
2011Q4	50	24	0.48	18	0.36	8	0.16
2012Q1	56	38	0.68	12	0.21	6	0.11

Continued on next page

Table B3: Time series data of quarterly onion roles for MongoDB

quarter	Total Members	Core	Share	Regular	Share	Casual	Share
2012Q2	44	20	0.45	18	0.41	6	0.14
2012Q3	64	42	0.66	16	0.25	6	0.09
2012Q4	54	36	0.67	12	0.22	6	0.11
2013Q1	66	44	0.67	16	0.24	6	0.09
2013Q2	56	38	0.68	12	0.21	6	0.11
2013Q3	58	32	0.55	18	0.31	8	0.14
2013Q4	74	38	0.51	26	0.35	10	0.14
2014Q1	68	42	0.62	18	0.26	8	0.12
2014Q2	52	32	0.62	14	0.27	6	0.12
2014Q3	90	52	0.58	28	0.31	10	0.11
2014Q4	66	28	0.42	28	0.42	10	0.15
2015Q1	52	30	0.58	16	0.31	6	0.12
2015Q2	46	26	0.57	14	0.30	6	0.13
2015Q3	30	16	0.53	10	0.33	4	0.13
2015Q4	38	26	0.68	8	0.21	4	0.11
2016Q1	34	24	0.71	6	0.18	4	0.12
2016Q2	32	24	0.75	6	0.19	2	0.06
2016Q3	32	24	0.75	6	0.19	2	0.06
2016Q4	22	16	0.73	4	0.18	2	0.09
2017Q1	20	14	0.70	4	0.20	2	0.10
2017Q2	34	24	0.71	6	0.18	4	0.12
2017Q3	24	16	0.67	6	0.25	2	0.08
2017Q4	34	26	0.76	6	0.18	2	0.06
2018Q1	32	18	0.56	10	0.31	4	0.12
2018Q2	22	14	0.64	6	0.27	2	0.09
2018Q3	20	14	0.70	4	0.20	2	0.10
2018Q4	18	12	0.67	4	0.22	2	0.11
2019Q1	22	16	0.73	4	0.18	2	0.09
2019Q2	22	16	0.73	4	0.18	2	0.09
2019Q3	18	12	0.67	4	0.22	2	0.11
2019Q4	32	24	0.75	6	0.19	2	0.06
2020Q1	24	18	0.75	4	0.17	2	0.08
2020Q2	24	18	0.75	4	0.17	2	0.08
2020Q3	16	12	0.75	2	0.12	2	0.12
2020Q4	16	12	0.75	2	0.12	2	0.12

Continued on next page

Table B3: Time series data of quarterly onion roles for MongoDB

quarter	Total Members	Core	Share	Regular	Share	Casual	Share
2021Q1	30	24	0.80	4	0.13	2	0.07
2021Q2	8	6	0.75	0	0.00	2	0.25
2021Q3	28	20	0.71	6	0.21	2	0.07
2021Q4	28	22	0.79	4	0.14	2	0.07
2022Q1	38	28	0.74	6	0.16	4	0.11
2022Q2	20	14	0.70	4	0.20	2	0.10
2022Q3	26	16	0.62	6	0.23	4	0.15
2022Q4	18	8	0.44	6	0.33	4	0.22

Table B4: Time series data of monthly organizational knowledge concentration for MongoDB

date	Total Contributions	Contributions by MongoDB	Share
2007-10-01	21	0	0.00
2007-11-01	67	1	0.01
2007-12-01	25	0	0.00
2008-01-01	12	1	0.08
2008-02-01	53	10	0.19
2008-03-01	79	0	0.00
2008-04-01	19	0	0.00
2008-05-01	6	0	0.00
2008-06-01	58	8	0.14
2008-07-01	60	10	0.17
2008-08-01	46	4	0.09
2008-09-01	47	11	0.23
2008-10-01	41	2	0.05
2008-11-01	72	0	0.00
2008-12-01	325	99	0.30
2009-01-01	605	357	0.59
2009-02-01	723	478	0.66
2009-03-01	321	279	0.87
2009-04-01	405	380	0.94
2009-05-01	455	422	0.93

Continued on next page

Table B4: Time series data of monthly organizational knowledge concentration for MongoDB

date	Total Contributions	Contributions by MongoDB	Share
2009-06-01	140	96	0.69
2009-07-01	169	134	0.79
2009-08-01	300	231	0.77
2009-09-01	300	217	0.72
2009-10-01	411	284	0.69
2009-11-01	289	180	0.62
2009-12-01	475	353	0.74
2010-01-01	428	319	0.75
2010-02-01	689	517	0.75
2010-03-01	550	433	0.79
2010-04-01	697	379	0.54
2010-05-01	599	444	0.74
2010-06-01	642	560	0.87
2010-07-01	1153	1011	0.88
2010-08-01	992	885	0.89
2010-09-01	458	406	0.89
2010-10-01	542	377	0.70
2010-11-01	548	455	0.83
2010-12-01	633	583	0.92
2011-01-01	540	418	0.77
2011-02-01	431	230	0.53
2011-03-01	733	451	0.62
2011-04-01	676	378	0.56
2011-05-01	727	433	0.60
2011-06-01	618	330	0.53
2011-07-01	649	346	0.53
2011-08-01	595	266	0.45
2011-09-01	594	222	0.37
2011-10-01	629	320	0.51
2011-11-01	509	254	0.50
2011-12-01	776	420	0.54
2012-01-01	606	265	0.44
2012-02-01	681	441	0.65
2012-03-01	653	354	0.54

Continued on next page

Table B4: Time series data of monthly organizational knowledge concentration for MongoDB

date	Total Contributions	Contributions by MongoDB	Share
2012-04-01	684	393	0.57
2012-05-01	675	296	0.44
2012-06-01	571	268	0.47
2012-07-01	596	246	0.41
2012-08-01	505	157	0.31
2012-09-01	519	106	0.20
2012-10-01	692	256	0.37
2012-11-01	655	317	0.48
2012-12-01	761	344	0.45
2013-01-01	550	232	0.42
2013-02-01	455	179	0.39
2013-03-01	483	196	0.41
2013-04-01	548	234	0.43
2013-05-01	613	259	0.42
2013-06-01	408	157	0.38
2013-07-01	658	225	0.34
2013-08-01	669	231	0.35
2013-09-01	497	199	0.40
2013-10-01	963	420	0.44
2013-11-01	766	208	0.27
2013-12-01	695	309	0.44
2014-01-01	563	311	0.55
2014-02-01	597	273	0.46
2014-03-01	922	462	0.50
2014-04-01	742	357	0.48
2014-05-01	675	331	0.49
2014-06-01	657	354	0.54
2014-07-01	675	303	0.45
2014-08-01	722	300	0.42
2014-09-01	914	383	0.42
2014-10-01	1244	670	0.54
2014-11-01	873	438	0.50
2014-12-01	959	368	0.38
2015-01-01	866	466	0.54

Continued on next page

Table B4: Time series data of monthly organizational knowledge concentration for MongoDB

date	Total Contributions	Contributions by MongoDB	Share
2015-02-01	723	332	0.46
2015-03-01	922	393	0.43
2015-04-01	1086	511	0.47
2015-05-01	913	437	0.48
2015-06-01	692	421	0.61
2015-07-01	863	435	0.50
2015-08-01	1050	585	0.56
2015-09-01	915	518	0.57
2015-10-01	712	438	0.62
2015-11-01	817	422	0.52
2015-12-01	791	409	0.52
2016-01-01	619	334	0.54
2016-02-01	829	499	0.60
2016-03-01	827	437	0.53
2016-04-01	802	518	0.65
2016-05-01	435	311	0.71
2016-06-01	509	412	0.81
2016-07-01	629	481	0.76
2016-08-01	635	466	0.73
2016-09-01	539	403	0.75
2016-10-01	340	267	0.79
2016-11-01	402	323	0.80
2016-12-01	433	332	0.77
2017-01-01	384	288	0.75
2017-02-01	324	260	0.80
2017-03-01	479	366	0.76
2017-04-01	501	376	0.75
2017-05-01	383	304	0.79
2017-06-01	603	468	0.78
2017-07-01	493	399	0.81
2017-08-01	551	418	0.76
2017-09-01	434	333	0.77
2017-10-01	486	394	0.81
2017-11-01	530	405	0.76

Continued on next page

Table B4: Time series data of monthly organizational knowledge concentration for MongoDB

date	Total Contributions	Contributions by MongoDB	Share
2017-12-01	374	273	0.73
2018-01-01	478	370	0.77
2018-02-01	551	428	0.78
2018-03-01	623	505	0.81
2018-04-01	591	469	0.79
2018-05-01	731	526	0.72
2018-06-01	752	591	0.79
2018-07-01	550	405	0.74
2018-08-01	565	438	0.78
2018-09-01	499	385	0.77
2018-10-01	484	376	0.78
2018-11-01	481	377	0.78
2018-12-01	526	391	0.74
2019-01-01	524	399	0.76
2019-02-01	457	320	0.70
2019-03-01	480	366	0.76
2019-04-01	544	400	0.74
2019-05-01	561	407	0.73
2019-06-01	757	533	0.70
2019-07-01	566	406	0.72
2019-08-01	519	384	0.74
2019-09-01	500	439	0.88
2019-10-01	646	552	0.85
2019-11-01	546	481	0.88
2019-12-01	656	594	0.91
2020-01-01	888	751	0.85
2020-02-01	741	561	0.76
2020-03-01	977	709	0.73
2020-04-01	941	707	0.75
2020-05-01	858	680	0.79
2020-06-01	710	510	0.72
2020-07-01	678	534	0.79
2020-08-01	692	521	0.75
2020-09-01	623	481	0.77

Continued on next page

Table B4: Time series data of monthly organizational knowledge concentration for MongoDB

date	Total Contributions	Contributions by MongoDB	Share
2020-10-01	632	476	0.75
2020-11-01	555	415	0.75
2020-12-01	401	308	0.77
2021-01-01	533	419	0.79
2021-02-01	697	529	0.76
2021-03-01	785	615	0.78
2021-04-01	762	583	0.77
2021-05-01	1153	854	0.74
2021-06-01	1202	971	0.81
2021-07-01	804	632	0.79
2021-08-01	738	584	0.79
2021-09-01	747	596	0.80
2021-10-01	1047	826	0.79
2021-11-01	787	600	0.76
2021-12-01	744	661	0.89
2022-01-01	774	648	0.84
2022-02-01	754	603	0.80
2022-03-01	965	766	0.79
2022-04-01	989	744	0.75
2022-05-01	877	703	0.80
2022-06-01	795	663	0.83
2022-07-01	598	513	0.86
2022-08-01	829	683	0.82
2022-09-01	833	670	0.80
2022-10-01	742	611	0.82
2022-11-01	796	629	0.79
2022-12-01	698	577	0.83

B.2 Elasticsearch

B.2.1 Community Activity

Table B5: Time series data of monthly community activity for Elasticsearch

date	Commits	Seasonal	Trend	Residual	Issues	Seasonal	Trend	Residual
2010-02-01	118	7.0	140.9	-29.9	45	6.9	35.3	2.8
2010-03-01	131	-24.4	143.6	11.8	55	17.9	38.6	-1.5
2010-04-01	163	8.0	146.2	8.8	55	0.7	41.9	12.3
2010-05-01	192	42.7	148.6	0.6	44	4.5	45.2	-5.7
2010-06-01	158	37.5	151.0	-30.5	44	11.1	48.6	-15.6
2010-07-01	154	11.7	153.3	-11.0	41	-10.1	51.9	-0.7
2010-08-01	211	36.1	155.6	19.4	60	10.1	55.1	-5.2
2010-09-01	132	-52.5	157.8	26.7	48	-14.1	58.4	3.7
2010-10-01	184	-46.5	160.0	70.5	72	2.6	61.7	7.7
2010-11-01	131	-19.8	162.3	-11.4	80	3.6	65.0	11.5
2010-12-01	132	7.0	164.5	-39.5	40	-26.7	68.3	-1.7
2011-01-01	171	11.9	166.5	-7.4	76	-0.1	71.9	4.3
2011-02-01	176	-0.5	167.5	8.9	68	5.4	74.9	-12.3
2011-03-01	147	-21.1	168.3	-0.2	92	11.5	77.4	3.1
2011-04-01	163	-13.6	168.8	7.8	74	1.1	79.3	-6.3
2011-05-01	191	27.0	169.2	-5.2	90	8.7	80.6	0.7
2011-06-01	124	-0.0	169.5	-45.5	96	0.4	81.6	14.1
2011-07-01	242	25.6	169.9	46.5	106	-4.1	82.2	27.9
2011-08-01	218	45.2	170.5	2.4	105	10.1	82.7	12.2
2011-09-01	185	4.8	171.2	8.9	79	-8.2	83.0	4.2
2011-10-01	98	-41.8	171.6	-31.8	62	-5.5	83.0	-15.5
2011-11-01	198	11.8	171.0	15.2	81	3.4	82.7	-5.0
2011-12-01	170	-33.3	169.3	34.0	66	-23.8	82.0	7.9
2012-01-01	168	14.8	166.4	-13.2	74	7.4	81.1	-14.5
2012-02-01	191	-12.3	163.1	40.2	96	2.6	80.1	13.3
2012-03-01	125	-16.7	160.1	-18.3	77	4.4	79.2	-6.5
2012-04-01	122	-30.6	157.6	-5.0	69	2.6	78.6	-12.2
2012-05-01	156	10.7	156.1	-10.7	98	11.6	78.4	7.9
2012-06-01	223	-46.6	155.5	114.1	77	-11.3	79.0	9.2
2012-07-01	133	40.5	156.0	-63.4	57	5.2	80.4	-28.6
2012-08-01	139	55.7	157.8	-74.5	97	11.5	82.5	3.0
2012-09-01	139	66.4	161.9	-89.3	73	-1.0	85.5	-11.5
2012-10-01	69	-27.6	169.1	-72.5	76	-14.6	89.4	1.2
2012-11-01	261	46.2	180.0	34.8	79	4.6	94.2	-19.8
2012-12-01	152	-73.4	195.0	30.4	61	-20.9	100.0	-18.0

Continued on next page

Table B5: Time series data of monthly community activity for Elasticsearch

date	Commits	Seasonal	Trend	Residual	Issues	Seasonal	Trend	Residual
2013-01-01	156	10.2	213.6	-67.8	93	11.6	107.0	-25.6
2013-02-01	180	-29.6	235.1	-25.4	105	-0.5	115.5	-10.0
2013-03-01	158	-14.6	259.7	-87.0	122	-4.3	125.8	0.5
2013-04-01	138	-42.5	288.6	-108.1	125	6.1	138.3	-19.5
2013-05-01	334	-3.9	322.4	15.4	166	13.7	153.1	-0.8
2013-06-01	310	-92.4	360.6	41.8	145	-24.0	169.6	-0.7
2013-07-01	411	48.5	402.3	-39.9	148	14.7	187.4	-54.2
2013-08-01	546	67.7	445.8	32.5	180	13.9	205.8	-39.7
2013-09-01	625	131.7	489.5	3.8	212	7.3	224.2	-19.5
2013-10-01	406	-0.9	532.8	-125.9	224	-23.3	242.5	4.8
2013-11-01	516	84.3	574.9	-143.1	271	8.2	260.5	2.3
2013-12-01	490	-110.3	614.9	-14.6	270	-19.2	278.4	10.8
2014-01-01	926	3.7	653.1	269.1	403	12.9	296.0	94.0
2014-02-01	677	-88.5	689.9	75.6	324	-15.9	313.0	26.9
2014-03-01	893	-3.5	723.9	172.6	329	-14.7	328.8	14.9
2014-04-01	756	-46.9	753.5	49.4	367	19.2	343.1	4.7
2014-05-01	696	-15.4	777.5	-66.1	362	11.0	355.4	-4.4
2014-06-01	490	-139.6	795.8	-166.2	294	-34.1	365.9	-37.8
2014-07-01	1025	5.2	810.0	209.7	457	22.4	375.4	59.2
2014-08-01	899	59.7	822.7	16.6	409	20.4	384.8	3.9
2014-09-01	1047	190.2	835.5	21.3	415	8.2	394.7	12.0
2014-10-01	1029	52.2	850.2	126.7	372	-34.5	406.0	0.5
2014-11-01	1097	133.9	869.2	93.9	403	38.2	419.2	-54.4
2014-12-01	600	-100.2	894.0	-193.8	405	-18.0	434.2	-11.2
2015-01-01	594	45.0	923.2	-374.1	389	5.9	450.6	-67.5
2015-02-01	746	-121.5	954.7	-87.2	414	-15.7	467.7	-37.9
2015-03-01	788	4.6	987.4	-203.9	427	-23.9	485.1	-34.3
2015-04-01	1096	-46.4	1020.4	122.0	554	31.0	502.6	20.4
2015-05-01	1147	-40.0	1054.1	132.9	527	17.9	519.7	-10.5
2015-06-01	944	-141.3	1089.2	-3.9	526	-32.3	536.2	22.1
2015-07-01	996	-49.8	1123.5	-77.7	623	22.6	551.6	48.8
2015-08-01	1324	18.4	1154.5	151.2	641	26.6	565.4	49.1
2015-09-01	1585	160.2	1180.2	244.6	654	-10.6	576.6	87.9
2015-10-01	1399	45.0	1198.1	156.0	534	-46.8	584.7	-3.9
2015-11-01	1374	204.1	1205.5	-35.7	711	58.6	589.2	63.2

Continued on next page

Table B5: Time series data of monthly community activity for Elasticsearch

date	Commits	Seasonal	Trend	Residual	Issues	Seasonal	Trend	Residual
2015-12-01	1300	-65.5	1203.0	162.6	603	-21.2	590.1	34.1
2016-01-01	1419	90.4	1193.3	135.3	604	-18.1	588.1	34.0
2016-02-01	969	-102.1	1180.1	-109.0	538	-7.9	584.2	-38.3
2016-03-01	1447	-69.7	1165.7	351.0	592	-37.7	579.1	50.6
2016-04-01	908	-27.6	1151.4	-215.8	612	31.8	573.3	6.9
2016-05-01	963	-16.4	1137.0	-157.6	594	39.4	567.6	-13.0
2016-06-01	1063	-77.8	1122.4	18.4	530	-15.1	562.3	-17.2
2016-07-01	905	-120.9	1109.3	-83.3	504	20.1	557.5	-73.5
2016-08-01	844	22.8	1098.3	-277.1	564	26.1	553.1	-15.1
2016-09-01	1069	70.5	1089.8	-91.3	441	-28.4	549.1	-79.7
2016-10-01	1054	-37.4	1084.5	6.9	514	-54.7	545.6	23.2
2016-11-01	1460	204.1	1083.4	172.5	674	58.4	542.2	73.3
2016-12-01	1098	-45.6	1085.6	58.0	498	-27.6	538.8	-13.2
2017-01-01	1319	156.1	1088.9	74.0	498	-22.7	534.8	-14.0
2017-02-01	1161	-27.2	1092.4	95.8	655	21.7	530.0	103.3
2017-03-01	715	-87.0	1094.7	-292.7	450	-33.6	524.8	-41.2
2017-04-01	1238	-19.2	1096.2	161.1	559	10.6	518.8	29.6
2017-05-01	1128	42.2	1096.9	-11.1	595	50.0	511.7	33.3
2017-06-01	1142	-35.3	1095.9	81.4	523	-15.4	503.8	34.6
2017-07-01	980	-126.0	1092.9	13.2	503	27.8	496.3	-21.1
2017-08-01	1198	32.6	1089.6	75.8	491	11.5	490.9	-11.4
2017-09-01	957	-22.7	1087.7	-108.0	412	-47.0	489.1	-30.1
2017-10-01	759	-80.6	1087.5	-247.8	366	-18.4	491.8	-107.4
2017-11-01	1205	117.2	1089.5	-1.7	436	15.9	499.4	-79.3
2017-12-01	1051	-141.9	1094.2	98.7	455	-78.4	512.4	20.9
2018-01-01	1377	144.5	1101.7	130.9	450	-13.6	531.3	-67.7
2018-02-01	1103	50.3	1112.8	-60.1	413	72.3	556.0	-215.3
2018-03-01	967	-70.0	1128.6	-91.6	508	3.6	586.8	-82.4
2018-04-01	1035	55.5	1148.5	-169.0	563	6.2	623.8	-67.0
2018-05-01	1293	127.7	1173.1	-7.8	734	38.2	666.9	28.9
2018-06-01	1174	-37.6	1202.0	9.6	690	-23.7	714.4	-0.7
2018-07-01	1033	-6.6	1234.5	-194.9	818	91.6	763.8	-37.4
2018-08-01	1460	34.3	1269.7	155.9	802	-17.2	812.9	6.3
2018-09-01	1179	-51.2	1306.9	-76.8	840	-46.6	859.9	26.7
2018-10-01	1230	-41.9	1345.8	-73.9	904	34.3	903.8	-34.1

Continued on next page

Table B5: Time series data of monthly community activity for Elasticsearch

date	Commits	Seasonal	Trend	Residual	Issues	Seasonal	Trend	Residual
2018-11-01	1410	-37.0	1384.8	62.2	983	-72.5	944.3	111.2
2018-12-01	1105	-275.4	1422.4	-42.1	928	-150.4	980.8	97.7
2019-01-01	1416	65.6	1456.9	-106.5	1080	-8.7	1013.8	74.9
2019-02-01	1800	46.3	1486.0	267.7	1396	65.6	1043.4	287.0
2019-03-01	1622	25.3	1508.0	88.8	1128	74.0	1068.9	-15.0
2019-04-01	1757	120.1	1522.3	114.5	1037	35.9	1089.8	-88.7
2019-05-01	1896	136.3	1528.6	231.1	1070	-14.5	1105.9	-21.3
2019-06-01	1398	-47.3	1527.9	-82.7	1016	-27.9	1118.9	-75.0
2019-07-01	1739	95.2	1522.3	121.5	1269	142.3	1131.4	-4.7
2019-08-01	1509	-2.9	1512.8	-0.9	1138	-54.3	1145.1	47.1
2019-09-01	1616	-11.0	1500.8	126.2	1120	-9.5	1161.7	-32.2
2019-10-01	1796	68.4	1488.6	239.0	1450	97.9	1182.7	169.4
2019-11-01	1121	-138.4	1475.9	-216.5	952	-123.8	1207.1	-131.3
2019-12-01	862	-327.1	1462.2	-273.1	806	-180.9	1232.6	-245.7
2020-01-01	1406	2.8	1447.1	-43.9	1230	13.8	1256.5	-40.3
2020-02-01	1212	56.2	1430.5	-274.7	1201	65.9	1276.6	-141.5
2020-03-01	1600	95.5	1413.5	91.0	1553	116.1	1293.2	143.7
2020-04-01	1581	86.4	1397.4	97.2	1509	11.1	1307.5	190.4
2020-05-01	1493	51.5	1383.4	58.1	1334	-66.1	1320.4	79.7
2020-06-01	1375	-32.5	1373.3	34.2	1386	-19.1	1332.8	72.3
2020-07-01	1839	47.4	1367.6	424.0	1763	70.3	1344.7	348.0
2020-08-01	1166	-39.0	1363.9	-158.9	1197	-61.9	1354.0	-95.1
2020-09-01	1292	42.6	1358.1	-108.6	1341	35.7	1358.2	-52.8
2020-10-01	1362	119.5	1346.1	-103.6	1349	116.2	1354.8	-122.0
2020-11-01	1143	-80.6	1327.7	-104.1	1182	-93.4	1344.6	-69.2
2020-12-01	1125	-277.6	1305.5	97.1	1255	-168.7	1331.0	92.7
2021-01-01	1256	14.0	1282.4	-40.4	1356	40.0	1317.5	-1.6
2021-02-01	1376	49.0	1260.4	66.6	1396	57.5	1305.9	32.5
2021-03-01	1525	133.3	1239.8	151.9	1497	150.5	1296.1	50.4
2021-04-01	1475	16.0	1220.1	238.9	1424	-30.3	1286.9	167.4
2021-05-01	940	-33.5	1200.4	-226.9	979	-98.8	1277.3	-199.5
2021-06-01	1150	-17.2	1177.8	-10.7	1228	-3.4	1264.9	-33.5
2021-07-01	947	-4.5	1149.2	-197.7	1101	1.3	1247.3	-147.6
2021-08-01	1049	-75.1	1115.0	9.1	1180	-69.2	1225.0	24.2
2021-09-01	1299	78.1	1077.6	143.3	1445	68.8	1199.6	176.6

Continued on next page

Table B5: Time series data of monthly community activity for Elasticsearch

date	Commits	Seasonal	Trend	Residual	Issues	Seasonal	Trend	Residual
2021-10-01	1452	130.3	1039.0	282.8	1575	100.2	1172.7	302.1
2021-11-01	951	1.2	1001.0	-51.2	1073	-48.9	1145.3	-23.4
2021-12-01	754	-209.0	963.7	-0.7	975	-142.4	1116.5	0.9
2022-01-01	986	34.9	926.7	24.4	1170	69.2	1085.2	15.6
2022-02-01	984	46.8	889.7	47.5	1118	46.9	1051.1	20.0
2022-03-01	859	160.9	851.2	-153.1	1118	179.6	1010.5	-72.1
2022-04-01	529	-64.5	813.6	-220.1	727	-77.2	970.7	-166.5
2022-05-01	767	-130.1	776.0	121.1	946	-130.5	931.1	145.4
2022-06-01	772	4.3	738.6	29.1	943	15.4	891.5	36.1
2022-07-01	574	-83.2	701.3	-44.1	749	-85.2	851.9	-17.7
2022-08-01	651	-99.6	664.2	86.4	790	-70.9	812.4	48.5
2022-09-01	686	115.1	627.3	-56.5	810	107.4	772.8	-70.2
2022-10-01	539	133.0	590.7	-184.7	646	81.0	733.1	-168.1
2022-11-01	808	105.4	554.3	148.3	807	9.2	693.5	104.3
2022-12-01	423	-126.7	518.2	31.5	582	-106.1	654.0	34.1
2023-01-01	584	65.7	482.5	35.8	746	102.2	614.6	29.1

B.2.2 Community Structure

Table B6: Time series data of monthly joining and leaving members for Elasticsearch

date	Joining	Seasonal	Trend	Residual	Leaving	Seasonal	Trend	Residual
2010-02-01	1	-1.65	2.17	0.48	0.00	0.00	0.00	0.00
2010-09-01	2	-1.29	2.31	0.98	1.00	0.06	1.05	-0.11
2010-10-01	2	-0.51	2.45	0.05	1.00	0.38	1.23	-0.60
2010-12-01	1	-0.04	2.59	-1.56	1.00	-0.69	1.40	0.29
2011-01-01	2	-0.21	2.73	-0.52	3.00	0.51	1.57	0.92
2011-02-01	3	-0.54	2.87	0.67	2.00	0.01	1.73	0.26
2011-03-01	1	0.07	3.00	-2.07	1.00	1.27	1.89	-2.17
2011-04-01	4	-0.88	3.13	1.75	1.00	-0.93	2.05	-0.12
2011-05-01	5	1.60	3.26	0.14	4.00	0.97	2.20	0.83
2011-06-01	2	0.71	3.38	-2.09	1.00	-0.55	2.35	-0.79
2011-07-01	5	-0.31	3.51	1.80	4.00	0.56	2.49	0.96

Continued on next page

Table B6: Time series data of monthly joining and leaving members for Elasticsearch

date	Joining	Seasonal	Trend	Residual	Leaving	Seasonal	Trend	Residual
2011-08-01	6	1.75	3.65	0.60	3.00	-0.71	2.63	1.08
2011-09-01	3	-1.25	3.74	0.51	3.00	-0.70	2.76	0.94
2011-10-01	3	-0.61	3.81	-0.19	3.00	-0.18	2.85	0.33
2011-11-01	5	0.19	3.86	0.95	5.00	0.88	2.92	1.21
2011-12-01	6	-0.05	3.90	2.15	2.00	-0.52	2.96	-0.44
2012-01-01	2	-1.50	3.90	-0.40	1.00	-0.85	2.97	-1.12
2012-02-01	2	-0.05	3.86	-1.81	3.00	0.57	2.94	-0.51
2012-03-01	5	0.33	3.79	0.88	5.00	0.68	2.90	1.42
2012-04-01	2	-0.82	3.72	-0.90	3.00	-0.68	2.86	0.81
2012-05-01	3	2.32	3.66	-2.98	2.00	2.15	2.84	-2.99
2012-06-01	9	0.38	3.65	4.98	4.00	-1.14	2.85	2.29
2012-07-01	2	-0.23	3.67	-1.44	2.00	0.23	2.88	-1.11
2012-08-01	5	-0.02	3.74	1.28	3.00	-0.45	2.92	0.52
2012-09-01	1	-0.72	3.85	-2.13	1.00	-0.47	2.97	-1.50
2012-10-01	2	0.19	3.98	-2.17	2.00	-0.39	3.01	-0.63
2012-11-01	3	0.97	4.13	-2.10	4.00	1.42	3.06	-0.49
2012-12-01	4	-0.05	4.31	-0.26	4.00	-0.24	3.14	1.09
2013-01-01	4	-2.78	4.56	2.22	2.00	-2.07	3.29	0.78
2013-02-01	5	0.29	4.91	-0.20	4.00	1.10	3.53	-0.63
2013-03-01	7	0.31	5.38	1.31	5.00	-0.11	3.84	1.27
2013-04-01	2	-0.60	5.98	-3.37	2.00	-0.36	4.23	-1.86
2013-05-01	10	2.55	6.73	0.72	5.00	2.77	4.72	-2.49
2013-06-01	5	0.31	7.63	-2.93	2.00	-1.55	5.32	-1.77
2013-07-01	5	-0.18	8.66	-3.48	4.00	-0.16	6.06	-1.90
2013-08-01	5	-1.53	9.82	-3.29	4.00	0.02	6.98	-3.00
2013-09-01	10	-0.14	11.09	-0.95	7.00	-0.32	8.07	-0.75
2013-10-01	13	1.12	12.43	-0.55	9.00	-0.57	9.30	0.27
2013-11-01	15	1.70	13.81	-0.52	11.00	1.89	10.60	-1.48
2013-12-01	14	-0.22	15.21	-0.99	9.00	0.27	11.91	-3.17
2014-01-01	11	-3.86	16.58	-1.72	7.00	-3.03	13.17	-3.14
2014-02-01	22	0.67	17.89	3.44	18.00	1.66	14.36	1.99
2014-03-01	23	0.01	19.09	3.90	17.00	-1.18	15.44	2.74
2014-04-01	22	-0.20	20.13	2.07	18.00	-0.14	16.41	1.73
2014-05-01	33	2.60	20.97	9.42	34.00	2.94	17.23	13.83
2014-06-01	19	-0.05	21.58	-2.52	15.00	-2.07	17.89	-0.81

Continued on next page

Table B6: Time series data of monthly joining and leaving members for Elasticsearch

date	Joining	Seasonal	Trend	Residual	Leaving	Seasonal	Trend	Residual
2014-07-01	25	-0.27	21.94	3.33	21.00	-0.72	18.35	3.37
2014-08-01	18	-3.05	22.07	-1.02	17.00	0.34	18.61	-1.95
2014-09-01	25	-0.41	22.03	3.38	20.00	-0.39	18.66	1.73
2014-10-01	26	2.05	21.88	2.06	17.00	-0.47	18.56	-1.09
2014-11-01	27	2.65	21.70	2.65	23.00	2.01	18.38	2.61
2014-12-01	22	0.37	21.55	0.08	20.00	1.83	18.22	-0.05
2015-01-01	14	-3.63	21.46	-3.83	16.00	-2.45	18.14	0.31
2015-02-01	20	0.84	21.43	-2.27	18.00	2.13	18.16	-2.29
2015-03-01	16	-0.48	21.42	-4.94	11.00	-2.38	18.28	-4.89
2015-04-01	21	1.82	21.42	-2.24	17.00	0.85	18.45	-2.30
2015-05-01	16	1.14	21.42	-6.55	14.00	1.26	18.64	-5.91
2015-06-01	24	-0.22	21.42	2.80	16.00	-1.67	18.84	-1.17
2015-07-01	25	-1.57	21.46	5.11	18.00	-1.63	19.05	0.58
2015-08-01	20	-3.00	21.57	1.43	27.00	0.54	19.28	7.18
2015-09-01	24	-1.67	21.74	3.93	22.00	-1.48	19.54	3.94
2015-10-01	27	1.87	21.91	3.22	21.00	-0.53	19.80	1.72
2015-11-01	24	3.17	22.01	-1.18	21.00	2.12	20.01	-1.13
2015-12-01	21	1.73	21.96	-2.69	25.00	2.68	20.11	2.21
2016-01-01	22	-2.14	21.74	2.40	18.00	0.11	20.06	-2.17
2016-02-01	22	0.77	21.40	-0.17	25.00	2.19	19.90	2.91
2016-03-01	19	0.44	21.04	-2.48	17.00	-2.14	19.70	-0.56
2016-04-01	27	2.84	20.72	3.44	23.00	1.45	19.52	2.03
2016-05-01	19	-0.95	20.51	-0.57	13.00	-1.39	19.39	-5.01
2016-06-01	23	-0.93	20.42	3.51	22.00	-1.54	19.33	4.20
2016-07-01	12	-3.50	20.41	-4.91	14.00	-2.83	19.34	-2.51
2016-08-01	18	-2.07	20.43	-0.36	16.00	1.02	19.39	-4.42
2016-09-01	9	-2.20	20.43	-9.23	11.00	-2.50	19.45	-5.96
2016-10-01	18	1.37	20.37	-3.73	19.00	-0.19	19.50	-0.31
2016-11-01	26	2.21	20.25	3.54	23.00	1.59	19.56	1.84
2016-12-01	30	1.48	20.14	8.38	28.00	1.26	19.64	7.10
2017-01-01	21	0.75	20.04	0.21	25.00	3.12	19.73	2.15
2017-02-01	22	0.75	19.94	1.31	20.00	2.24	19.78	-2.03
2017-03-01	26	2.96	19.82	3.21	20.00	0.72	19.74	-0.46
2017-04-01	24	2.40	19.65	1.95	23.00	1.11	19.59	2.30
2017-05-01	18	-0.98	19.39	-0.41	20.00	-3.04	19.36	3.68

Continued on next page

Table B6: Time series data of monthly joining and leaving members for Elasticsearch

date	Joining	Seasonal	Trend	Residual	Leaving	Seasonal	Trend	Residual
2017-06-01	14	-2.40	19.03	-2.64	16.00	-1.39	19.04	-1.65
2017-07-01	12	-5.03	18.58	-1.55	16.00	-3.15	18.65	0.51
2017-08-01	16	-1.87	18.07	-0.20	19.00	-0.20	18.23	0.97
2017-09-01	19	-1.84	17.58	3.26	17.00	-3.26	17.82	2.44
2017-10-01	18	0.58	17.16	0.26	16.00	0.28	17.45	-1.72
2017-11-01	15	1.43	16.84	-3.27	16.00	1.32	17.09	-2.41
2017-12-01	13	-0.21	16.60	-3.39	8.00	-0.87	16.74	-7.87
2018-01-01	18	3.20	16.45	-1.65	23.00	4.95	16.40	1.65
2018-02-01	18	0.41	16.41	1.18	21.00	1.55	16.13	3.32
2018-03-01	22	5.53	16.45	0.02	20.00	4.18	15.97	-0.15
2018-04-01	16	0.82	16.58	-1.40	15.00	-0.14	15.92	-0.78
2018-05-01	18	-0.01	16.78	1.23	15.00	-2.94	15.94	1.99
2018-06-01	10	-3.12	17.06	-3.94	12.00	-1.90	16.04	-2.14
2018-07-01	12	-4.02	17.38	-1.36	11.00	-1.44	16.18	-3.73
2018-08-01	18	-2.42	17.73	2.69	17.00	-1.38	16.30	2.08
2018-09-01	17	0.48	18.08	-1.56	11.00	-2.09	16.40	-3.31
2018-10-01	20	0.77	18.40	0.83	19.00	1.01	16.49	1.50
2018-11-01	19	-0.11	18.70	0.41	19.00	0.27	16.60	2.13
2018-12-01	17	-3.17	19.00	1.17	14.00	-2.94	16.77	0.17
2019-01-01	27	4.43	19.31	3.27	24.00	4.45	16.99	2.56
2019-02-01	17	-0.38	19.60	-2.22	14.00	1.67	17.24	-4.90
2019-03-01	28	5.58	19.83	2.59	28.00	5.62	17.48	4.90
2019-04-01	17	-0.53	19.97	-2.44	14.00	-1.44	17.69	-2.25
2019-05-01	20	0.28	20.03	-0.32	9.00	-2.32	17.87	-6.55
2019-06-01	19	-1.86	20.05	0.81	19.00	-2.31	18.01	3.30
2019-07-01	20	-2.07	20.04	2.04	22.00	-1.07	18.14	4.93
2019-08-01	14	-2.22	20.01	-3.79	13.00	-0.91	18.27	-4.36
2019-09-01	27	0.91	19.99	6.10	21.00	-1.42	18.41	4.01
2019-10-01	20	1.77	19.99	-1.75	19.00	2.62	18.55	-2.17
2019-11-01	22	-1.67	19.96	3.71	19.00	-1.13	18.67	1.47
2019-12-01	9	-3.89	19.89	-7.00	16.00	-3.86	18.74	1.13
2020-01-01	25	5.19	19.72	0.09	20.00	4.90	18.74	-3.64
2020-02-01	20	-1.20	19.45	1.74	23.00	0.77	18.67	3.56
2020-03-01	24	4.29	19.10	0.61	24.00	5.41	18.52	0.07
2020-04-01	19	-1.02	18.69	1.32	17.00	-1.93	18.29	0.64

Continued on next page

Table B6: Time series data of monthly joining and leaving members for Elasticsearch

date	Joining	Seasonal	Trend	Residual	Leaving	Seasonal	Trend	Residual
2020-05-01	20	-0.18	18.24	1.94	19.00	-0.83	18.02	1.81
2020-06-01	20	-1.17	17.76	3.41	13.00	-3.19	17.71	-1.52
2020-07-01	18	-1.51	17.25	2.26	20.00	-1.13	17.35	3.77
2020-08-01	15	-0.97	16.73	-0.76	16.00	-0.44	16.98	-0.54
2020-09-01	14	0.38	16.20	-2.58	15.00	-0.76	16.62	-0.86
2020-10-01	19	2.10	15.65	1.25	22.00	4.07	16.27	1.66
2020-11-01	7	-1.81	15.10	-6.29	10.00	-2.30	15.93	-3.63
2020-12-01	16	-3.39	14.56	4.83	15.00	-5.05	15.62	4.44
2021-01-01	18	5.92	14.07	-1.99	14.00	6.05	15.35	-7.40
2021-02-01	10	-1.84	13.66	-1.82	17.00	-0.32	15.11	2.21
2021-03-01	13	3.11	13.34	-3.45	16.00	5.28	14.91	-4.19
2021-04-01	13	-1.56	13.11	1.44	13.00	-2.34	14.78	0.56
2021-05-01	10	-0.56	12.98	-2.42	16.00	0.95	14.72	0.33
2021-06-01	15	-0.95	12.91	3.04	14.00	-4.59	14.73	3.86
2021-07-01	11	-1.11	12.89	-0.78	9.00	-1.34	14.78	-4.44
2021-08-01	14	0.38	12.88	0.74	17.00	0.16	14.86	1.97
2021-09-01	9	0.02	12.86	-3.88	14.00	-0.20	14.89	-0.69
2021-10-01	17	2.28	12.80	1.92	20.00	5.59	14.90	-0.49
2021-11-01	12	-2.10	12.74	1.37	13.00	-3.60	14.90	1.70
2021-12-01	10	-2.68	12.66	0.02	5.00	-6.23	14.90	-3.66
2022-01-01	20	6.52	12.56	0.92	29.00	7.39	14.89	6.72
2022-02-01	11	-2.42	12.43	0.99	11.00	-1.41	14.89	-2.48
2022-03-01	16	1.83	12.29	1.88	22.00	4.92	14.89	2.19
2022-04-01	9	-2.00	12.15	-1.16	12.00	-2.62	14.88	-0.27
2022-05-01	12	-1.00	12.00	0.99	18.00	3.01	14.87	0.11
2022-06-01	7	-0.97	11.85	-3.88	6.00	-6.09	14.86	-2.76
2022-07-01	10	-0.92	11.69	-0.77	0.00	0.00	0.00	0.00
2022-08-01	14	1.99	11.52	0.48	0.00	0.00	0.00	0.00
2022-09-01	14	-0.42	11.35	3.07	0.00	0.00	0.00	0.00
2022-10-01	12	2.47	11.19	-1.66	0.00	0.00	0.00	0.00
2022-11-01	10	-2.22	11.02	1.20	0.00	0.00	0.00	0.00
2022-12-01	8	-1.88	10.85	-0.97	0.00	0.00	0.00	0.00

Table B7: Time series data of quarterly onion roles for Elasticsearch

quarter	Total Members	Core	Share	Regular	Share	Casual	Share
2010Q1	18	2	0.11	8	0.44	8	0.44
2010Q2	36	2	0.06	18	0.50	16	0.44
2010Q3	50	8	0.16	24	0.48	18	0.36
2010Q4	86	21	0.24	42	0.49	23	0.27
2011Q1	128	46	0.36	56	0.44	26	0.20
2011Q2	170	64	0.38	78	0.46	28	0.16
2011Q3	200	81	0.41	86	0.43	33	0.17
2011Q4	182	98	0.54	62	0.34	22	0.12
2012Q1	220	118	0.54	74	0.34	28	0.13
2012Q2	216	116	0.54	74	0.34	26	0.12
2012Q3	210	116	0.55	67	0.32	27	0.13
2012Q4	234	142	0.61	64	0.27	28	0.12
2013Q1	316	189	0.60	91	0.29	36	0.11
2013Q2	446	269	0.60	128	0.29	49	0.11
2013Q3	510	293	0.57	154	0.30	63	0.12
2013Q4	524	236	0.45	209	0.40	79	0.15
2014Q1	768	388	0.51	270	0.35	110	0.14
2014Q2	846	459	0.54	284	0.34	103	0.12
2014Q3	910	467	0.51	315	0.35	128	0.14
2014Q4	1012	568	0.56	321	0.32	123	0.12
2015Q1	1134	673	0.59	333	0.29	128	0.11
2015Q2	1154	625	0.54	368	0.32	161	0.14
2015Q3	1038	451	0.43	401	0.39	186	0.18
2015Q4	1274	678	0.53	414	0.32	182	0.14
2016Q1	1130	571	0.51	393	0.35	166	0.15
2016Q2	1158	612	0.53	370	0.32	176	0.15
2016Q3	1038	571	0.55	319	0.31	148	0.14
2016Q4	1274	690	0.54	413	0.32	171	0.13
2017Q1	1242	637	0.51	442	0.36	163	0.13
2017Q2	1128	579	0.51	382	0.34	167	0.15
2017Q3	1106	602	0.54	360	0.33	144	0.13
2017Q4	1008	567	0.56	317	0.31	124	0.12
2018Q1	1084	584	0.54	360	0.33	140	0.13
2018Q2	1098	569	0.52	336	0.31	193	0.18
2018Q3	1116	517	0.46	364	0.33	235	0.21

Continued on next page

Table B7: Time series data of quarterly onion roles for Elasticsearch

quarter	Total Members	Core	Share	Regular	Share	Casual	Share
2018Q4	1144	463	0.40	411	0.36	270	0.24
2019Q1	1112	423	0.38	376	0.34	313	0.28
2019Q2	1132	482	0.43	357	0.32	293	0.26
2019Q3	1156	500	0.43	355	0.31	301	0.26
2019Q4	1154	455	0.39	400	0.35	299	0.26
2020Q1	1112	392	0.35	384	0.35	336	0.30
2020Q2	1076	413	0.38	337	0.31	326	0.30
2020Q3	996	348	0.35	319	0.32	329	0.33
2020Q4	1000	405	0.41	297	0.30	298	0.30
2021Q1	888	307	0.35	303	0.34	278	0.31
2021Q2	930	347	0.37	290	0.31	293	0.32
2021Q3	824	299	0.36	269	0.33	256	0.31
2021Q4	778	271	0.35	264	0.34	243	0.31
2022Q1	916	330	0.36	311	0.34	275	0.30
2022Q2	806	310	0.38	276	0.34	220	0.27
2022Q3	788	321	0.41	256	0.32	211	0.27
2022Q4	706	296	0.42	224	0.32	186	0.26

Table B8: Time series data of monthly organizational knowledge concentration for Elasticsearch

date	Total Contributions	Contributions by Elasticsearch	Share
2010-02-01	163	0	0.00
2010-03-01	191	0	0.00
2010-04-01	214	0	0.00
2010-05-01	235	0	0.00
2010-06-01	205	0	0.00
2010-07-01	192	0	0.00
2010-08-01	272	0	0.00
2010-09-01	179	0	0.00
2010-10-01	256	0	0.00
2010-11-01	214	0	0.00
2010-12-01	171	0	0.00

Continued on next page

Table B8: Time series data of monthly organizational knowledge concentration for Elasticsearch

date	Total Contributions	Contributions by Elasticsearch	Share
2011-01-01	249	0	0.00
2011-02-01	246	0	0.00
2011-03-01	234	0	0.00
2011-04-01	236	0	0.00
2011-05-01	287	0	0.00
2011-06-01	215	0	0.00
2011-07-01	355	0	0.00
2011-08-01	326	0	0.00
2011-09-01	253	0	0.00
2011-10-01	164	0	0.00
2011-11-01	277	0	0.00
2011-12-01	235	0	0.00
2012-01-01	242	0	0.00
2012-02-01	286	0	0.00
2012-03-01	202	0	0.00
2012-04-01	191	0	0.00
2012-05-01	260	0	0.00
2012-06-01	299	0	0.00
2012-07-01	191	0	0.00
2012-08-01	236	2	0.01
2012-09-01	211	2	0.01
2012-10-01	239	1	0.00
2012-11-01	250	5	0.02
2012-12-01	210	0	0.00
2013-01-01	252	0	0.00
2013-02-01	295	5	0.02
2013-03-01	275	1	0.00
2013-04-01	276	0	0.00
2013-05-01	498	1	0.00
2013-06-01	456	2	0.00
2013-07-01	567	7	0.01
2013-08-01	726	42	0.06
2013-09-01	846	24	0.03
2013-10-01	629	15	0.02

Continued on next page

Table B8: Time series data of monthly organizational knowledge concentration for Elasticsearch

date	Total Contributions	Contributions by Elasticsearch	Share
2013-11-01	787	5	0.01
2013-12-01	793	4	0.01
2014-01-01	1333	22	0.02
2014-02-01	1042	23	0.02
2014-03-01	1240	24	0.02
2014-04-01	1108	47	0.04
2014-05-01	1094	55	0.05
2014-06-01	812	17	0.02
2014-07-01	1540	69	0.04
2014-08-01	1252	121	0.10
2014-09-01	1404	123	0.09
2014-10-01	1347	167	0.12
2014-11-01	1501	99	0.07
2014-12-01	1016	37	0.04
2015-01-01	971	71	0.07
2015-02-01	1214	85	0.07
2015-03-01	1174	118	0.10
2015-04-01	1611	132	0.08
2015-05-01	1673	194	0.12
2015-06-01	1486	155	0.10
2015-07-01	1613	226	0.14
2015-08-01	1997	261	0.13
2015-09-01	2267	394	0.17
2015-10-01	1894	395	0.21
2015-11-01	2107	369	0.18
2015-12-01	1885	394	0.21
2016-01-01	2032	477	0.23
2016-02-01	1516	301	0.20
2016-03-01	1987	436	0.22
2016-04-01	1523	296	0.19
2016-05-01	1591	421	0.26
2016-06-01	1575	313	0.20
2016-07-01	1381	257	0.19
2016-08-01	1409	359	0.25

Continued on next page

Table B8: Time series data of monthly organizational knowledge concentration for Elasticsearch

date	Total Contributions	Contributions by Elasticsearch	Share
2016-09-01	1506	396	0.26
2016-10-01	1569	387	0.25
2016-11-01	2122	645	0.30
2016-12-01	1621	419	0.26
2017-01-01	1786	408	0.23
2017-02-01	1845	382	0.21
2017-03-01	1176	303	0.26
2017-04-01	1854	580	0.31
2017-05-01	1705	459	0.27
2017-06-01	1665	542	0.33
2017-07-01	1496	390	0.26
2017-08-01	1695	477	0.28
2017-09-01	1359	405	0.30
2017-10-01	1132	291	0.26
2017-11-01	1698	585	0.34
2017-12-01	1499	410	0.27
2018-01-01	1812	531	0.29
2018-02-01	1481	427	0.29
2018-03-01	1482	422	0.28
2018-04-01	1651	518	0.31
2018-05-01	2035	676	0.33
2018-06-01	1903	610	0.32
2018-07-01	1866	498	0.27
2018-08-01	2063	627	0.30
2018-09-01	2021	637	0.32
2018-10-01	2125	578	0.27
2018-11-01	2393	610	0.25
2018-12-01	2032	571	0.28
2019-01-01	2467	702	0.28
2019-02-01	3218	877	0.27
2019-03-01	2747	965	0.35
2019-04-01	2811	1173	0.42
2019-05-01	2965	1467	0.49
2019-06-01	2421	821	0.34

Continued on next page

Table B8: Time series data of monthly organizational knowledge concentration for Elasticsearch

date	Total Contributions	Contributions by Elasticsearch	Share
2019-07-01	3000	1409	0.47
2019-08-01	2649	1111	0.42
2019-09-01	2740	1225	0.45
2019-10-01	3231	1268	0.39
2019-11-01	2085	848	0.41
2019-12-01	1666	695	0.42
2020-01-01	2634	1050	0.40
2020-02-01	2409	822	0.34
2020-03-01	3141	1199	0.38
2020-04-01	3081	1250	0.41
2020-05-01	2824	1153	0.41
2020-06-01	2765	1037	0.38
2020-07-01	3614	1280	0.35
2020-08-01	2360	709	0.30
2020-09-01	2632	746	0.28
2020-10-01	2716	864	0.32
2020-11-01	2324	654	0.28
2020-12-01	2367	624	0.26
2021-01-01	2612	695	0.27
2021-02-01	2783	881	0.32
2021-03-01	3037	826	0.27
2021-04-01	2883	807	0.28
2021-05-01	1904	484	0.25
2021-06-01	2368	667	0.28
2021-07-01	2055	551	0.27
2021-08-01	2233	673	0.30
2021-09-01	2740	802	0.29
2021-10-01	3028	1067	0.35
2021-11-01	2022	688	0.34
2021-12-01	1729	559	0.32
2022-01-01	2156	919	0.43
2022-02-01	2099	668	0.32
2022-03-01	1975	570	0.29
2022-04-01	1265	336	0.27

Continued on next page

Table B8: Time series data of monthly organizational knowledge concentration for Elasticsearch

date	Total Contributions	Contributions by Elasticsearch	Share
2022-05-01	1727	582	0.34
2022-06-01	1695	499	0.29
2022-07-01	1324	321	0.24
2022-08-01	1441	420	0.29
2022-09-01	1498	509	0.34
2022-10-01	1192	437	0.37
2022-11-01	1605	553	0.34
2022-12-01	1001	303	0.30

B.3 Redis

B.3.1 Community Activity

Table B9: Time series data of monthly community activity for Redis

date	Commits	Seasonal	Trend	Residual	Issues	Seasonal	Trend	Residual
2016-04-01	43	-10.7	14.5	39.2	0.0	0.0	0.0	0.0
2016-05-01	86	39.3	23.4	23.3	0.0	0.0	0.0	0.0
2016-06-01	37	13.0	32.4	-8.4	2.0	5.1	-9.3	6.1
2016-07-01	27	18.8	41.4	-33.2	0.0	11.7	-6.2	-5.5
2016-08-01	25	8.5	50.5	-34.0	3.0	15.5	-3.1	-9.3
2016-09-01	5	-75.9	59.6	21.3	0.0	-5.6	-0.1	5.7
2016-10-01	18	-32.9	68.7	-17.8	0.0	2.1	2.9	-5.0
2016-11-01	18	-9.3	77.8	-50.6	1.0	5.4	5.8	-10.2
2016-12-01	149	4.6	87.0	57.3	15.0	-1.6	8.7	7.9
2017-01-01	154	52.3	96.3	5.3	4.0	-5.0	11.5	-2.5
2017-02-01	85	-40.8	106.0	19.8	4.0	-17.8	14.3	7.5
2017-03-01	119	33.1	116.7	-30.8	21.0	-6.6	17.1	10.5
2017-04-01	78	-12.7	126.2	-35.4	7.0	-19.0	20.0	6.0
2017-05-01	134	20.2	133.9	-20.0	35.0	6.9	22.9	5.1
2017-06-01	177	3.0	140.0	34.0	27.0	2.0	25.2	-0.2
2017-07-01	238	38.0	145.0	55.0	52.0	19.1	27.2	5.7
2017-08-01	235	18.7	149.0	67.3	42.0	19.4	28.9	-6.3

Continued on next page

Table B9: Time series data of monthly community activity for Redis

date	Commits	Seasonal	Trend	Residual	Issues	Seasonal	Trend	Residual
2017-09-01	68	-59.9	151.8	-23.9	29.0	-2.2	30.5	0.7
2017-10-01	176	-23.9	153.5	46.4	34.0	4.4	32.2	-2.6
2017-11-01	213	-11.4	154.0	70.4	33.0	3.2	34.1	-4.4
2017-12-01	72	-1.7	153.0	-79.3	34.0	0.2	36.5	-2.7
2018-01-01	191	24.5	150.2	16.3	36.0	-14.2	39.3	10.9
2018-02-01	62	-20.9	145.9	-63.0	22.0	-8.8	42.7	-11.8
2018-03-01	246	23.5	140.7	81.8	19.0	-11.6	46.4	-15.8
2018-04-01	125	-13.0	135.5	2.4	22.0	-13.5	50.2	-14.6
2018-05-01	127	1.5	131.2	-5.7	47.0	-4.8	54.0	-2.2
2018-06-01	104	-5.6	127.5	-17.9	49.0	-0.0	57.8	-8.8
2018-07-01	193	57.5	124.3	11.2	92.0	26.0	61.6	4.5
2018-08-01	106	27.9	121.4	-43.4	121.0	21.7	65.4	33.9
2018-09-01	63	-45.5	118.9	-10.5	52.0	1.2	69.2	-18.5
2018-10-01	69	-14.7	116.7	-33.0	96.0	6.0	72.9	17.1
2018-11-01	130	-13.0	114.9	28.1	108.0	-0.4	76.5	31.9
2018-12-01	111	-8.8	114.2	5.6	63.0	2.1	79.8	-19.0
2019-01-01	92	-1.4	115.0	-21.6	53.0	-22.2	82.8	-7.6
2019-02-01	187	-4.4	116.9	74.5	82.0	-0.8	85.4	-2.7
2019-03-01	70	18.3	119.5	-67.7	72.0	-16.5	87.9	0.5
2019-04-01	68	-9.0	122.1	-45.1	92.0	-9.3	90.4	10.9
2019-05-01	127	-13.5	124.4	16.1	75.0	-14.2	93.0	-3.8
2019-06-01	103	-15.4	126.3	-8.0	85.0	-1.1	95.9	-9.9
2019-07-01	160	72.8	128.3	-41.0	120.0	31.0	99.1	-10.2
2019-08-01	222	30.5	130.1	61.4	118.0	23.5	102.3	-7.9
2019-09-01	144	-29.7	132.0	41.7	116.0	3.3	105.6	7.1
2019-10-01	150	-8.8	134.1	24.7	113.0	7.4	109.0	-3.4
2019-11-01	72	-17.5	135.8	-46.4	106.0	-4.0	112.4	-2.4
2019-12-01	156	-8.2	136.9	27.3	132.0	3.4	116.0	12.6
2020-01-01	93	-26.6	137.0	-17.3	80.0	-30.0	119.6	-9.6
2020-02-01	134	16.2	135.6	-17.8	152.0	5.8	123.1	23.2
2020-03-01	150	11.7	133.1	5.2	113.0	-12.9	126.2	-0.4
2020-04-01	206	-20.3	130.0	96.3	130.0	-6.5	129.0	7.5
2020-05-01	64	-11.4	127.2	-51.7	85.0	-13.7	131.0	-32.3
2020-06-01	97	-18.2	125.0	-9.8	139.0	4.5	132.4	2.0
2020-07-01	242	57.8	124.1	60.1	198.0	22.5	133.4	42.1

Continued on next page

Table B9: Time series data of monthly community activity for Redis

date	Commits	Seasonal	Trend	Residual	Issues	Seasonal	Trend	Residual
2020-08-01	99	34.3	124.3	-59.6	145.0	22.9	134.4	-12.3
2020-09-01	90	-24.4	125.6	-11.2	172.0	-3.3	135.6	39.8
2020-10-01	102	-10.2	127.9	-15.6	133.0	6.6	136.8	-10.4
2020-11-01	72	-7.6	131.1	-51.5	97.0	-7.4	138.0	-33.7
2020-12-01	106	9.3	135.3	-38.6	160.0	-1.7	139.0	22.8
2021-01-01	108	12.0	140.5	-44.6	100.0	-29.0	139.7	-10.7
2021-02-01	175	-26.9	147.1	54.8	132.0	13.6	140.4	-22.0
2021-03-01	163	23.4	154.5	-14.9	123.0	-0.0	141.1	-18.1
2021-04-01	105	-33.0	161.9	-23.9	128.0	-6.6	142.2	-7.6
2021-05-01	195	-14.4	169.1	40.3	179.0	-15.8	143.9	50.9
2021-06-01	179	-23.8	176.3	26.6	173.0	9.0	146.5	17.6
2021-07-01	240	40.0	183.0	17.0	134.0	16.0	149.4	-31.4
2021-08-01	238	35.8	189.0	13.2	163.0	23.7	152.5	-13.1
2021-09-01	169	-22.1	194.2	-3.1	114.0	-8.3	155.2	-32.9
2021-10-01	216	-15.6	198.3	33.3	172.0	5.0	157.5	9.5
2021-11-01	248	2.1	201.8	44.1	167.0	-12.0	159.7	19.3
2021-12-01	215	26.8	204.9	-16.6	145.0	-7.2	161.7	-9.6
2022-01-01	151	66.7	207.8	-123.5	162.0	-29.4	163.7	27.7
2022-02-01	191	-80.7	210.5	61.1	194.0	19.7	165.8	8.5
2022-03-01	263	38.6	213.8	10.6	194.0	13.2	167.5	13.3
2022-04-01	149	-49.7	220.4	-21.7	164.0	-7.8	168.8	2.9
2022-05-01	199	-16.0	227.2	-12.2	131.0	-15.5	170.0	-23.5
2022-06-01	189	-28.8	233.9	-16.1	174.0	14.0	171.1	-11.1
2022-07-01	225	20.1	240.4	-35.5	187.0	7.9	172.1	7.0
2022-08-01	290	37.9	247.0	5.1	216.0	25.6	173.1	17.3
2022-09-01	229	-21.9	253.7	-2.8	165.0	-15.5	174.0	6.5
2022-10-01	215	-21.6	260.5	-23.9	178.0	4.0	174.9	-0.9
2022-11-01	281	16.9	267.5	-3.4	163.0	-14.8	175.7	2.0
2022-12-01	341	44.4	274.6	22.0	158.0	-14.4	176.5	-4.2
2023-01-01	523	126.8	281.9	114.2	136.0	-28.6	177.3	-12.7
2023-02-01	78	-139.3	289.4	-72.1	0.0	0.0	0.0	0.0

B.3.2 Community Structure

Table B10: Time series data of monthly joining and leaving members for Redis

date	Joining	Seasonal	Trend	Residual	Leaving	Seasonal	Trend	Residual
2016-04-01	2.00	1.13	1.85	-0.98	0.00	0.00	0.00	0.00
2016-05-01	3.00	0.78	1.86	0.36	2.00	0.13	1.73	0.14
2016-07-01	2.00	-0.13	1.86	0.27	2.00	-0.02	1.75	0.27
2016-08-01	2.00	0.03	1.87	0.10	2.00	0.33	1.77	-0.10
2016-09-01	1.00	-1.04	1.87	0.17	0.00	0.00	0.00	0.00
2016-10-01	1.00	-0.33	1.87	-0.54	1.00	-0.56	1.79	-0.23
2016-11-01	2.00	-0.48	1.87	0.61	2.00	-0.67	1.82	0.86
2016-12-01	2.00	-0.24	1.87	0.36	1.00	-0.41	1.84	-0.43
2017-02-01	1.00	-1.05	1.87	0.17	1.00	0.59	1.86	-1.45
2017-03-01	3.00	0.84	1.87	0.29	3.00	0.48	1.88	0.64
2017-04-01	1.00	0.45	1.86	-1.32	2.00	0.30	1.91	-0.20
2017-05-01	2.00	-0.06	1.84	0.22	2.00	-0.13	1.93	0.21
2017-06-01	4.00	1.05	1.81	1.14	3.00	1.01	1.95	0.04
2017-07-01	2.00	0.13	1.80	0.07	2.00	-0.58	1.98	0.59
2017-08-01	2.00	0.07	1.81	0.12	2.00	-0.19	2.03	0.16
2017-10-01	1.00	0.62	1.84	-1.46	0.00	0.00	0.00	0.00
2017-11-01	1.00	-0.84	1.88	-0.04	1.00	-0.34	2.08	-0.74
2017-12-01	2.00	-0.29	1.92	0.36	2.00	-0.25	2.12	0.14
2018-01-01	1.00	-0.62	1.98	-0.36	2.00	-0.28	2.14	0.14
2018-02-01	1.00	0.12	2.05	-1.17	0.00	0.00	0.00	0.00
2018-03-01	1.00	-0.71	2.12	-0.41	1.00	0.04	2.15	-1.19
2018-04-01	2.00	-0.02	2.20	-0.18	2.00	-0.40	2.15	0.25
2018-05-01	5.00	0.63	2.29	2.08	5.00	0.61	2.14	2.25
2018-06-01	2.00	-0.23	2.39	-0.16	1.00	0.04	2.13	-1.17
2018-07-01	4.00	0.96	2.48	0.56	3.00	0.13	2.12	0.74
2018-09-01	1.00	-0.45	2.55	-1.10	0.00	0.00	0.00	0.00
2018-10-01	2.00	0.41	2.60	-1.00	2.00	0.09	2.11	-0.20
2018-11-01	6.00	0.93	2.61	2.45	4.00	1.33	2.10	0.57
2018-12-01	0.00	0.00	0.00	0.00	1.00	-0.28	2.07	-0.80
2019-01-01	2.00	-0.51	2.61	-0.11	1.00	-0.41	2.03	-0.63
2019-02-01	3.00	-0.33	2.59	0.73	2.00	-0.70	1.98	0.72
2019-03-01	1.00	-0.59	2.56	-0.98	1.00	-0.85	1.91	-0.06
2019-04-01	0.00	0.00	0.00	0.00	2.00	-0.08	1.84	0.24
2019-05-01	4.00	0.36	2.53	1.11	2.00	0.71	1.78	-0.49
2019-06-01	2.00	-0.45	2.50	-0.05	2.00	-0.48	1.74	0.74

Continued on next page

Table B10: Time series data of monthly joining and leaving members for Redis

date	Joining	Seasonal	Trend	Residual	Leaving	Seasonal	Trend	Residual
2019-07-01	1.00	-0.83	2.46	-0.63	0.00	0.00	0.00	0.00
2019-08-01	3.00	0.84	2.43	-0.27	2.00	0.62	1.72	-0.34
2019-09-01	2.00	-0.30	2.40	-0.10	2.00	-0.39	1.74	0.65
2019-10-01	2.00	0.75	2.40	-1.15	1.00	0.07	1.80	-0.87
2019-11-01	2.00	-0.97	2.41	0.56	2.00	0.34	1.88	-0.22
2019-12-01	3.00	0.79	2.44	-0.23	2.00	1.71	1.97	-1.67
2020-01-01	4.00	0.64	2.49	0.87	2.00	0.06	2.06	-0.12
2020-02-01	1.00	0.17	2.54	-1.72	1.00	-0.27	2.16	-0.88
2020-03-01	2.00	-0.74	2.62	0.12	1.00	-0.98	2.27	-0.29
2020-04-01	2.00	0.31	2.71	-1.02	1.00	-1.50	2.40	0.10
2020-05-01	4.00	0.11	2.82	1.08	3.00	-0.28	2.55	0.73
2020-06-01	4.00	-0.08	2.93	1.15	6.00	1.35	2.73	1.92
2020-07-01	2.00	-1.23	3.04	0.18	2.00	-1.00	2.91	0.09
2020-08-01	3.00	0.62	3.15	-0.77	3.00	-0.06	3.07	-0.02
2020-09-01	2.00	0.12	3.22	-1.34	2.00	-0.30	3.20	-0.90
2020-10-01	5.00	0.26	3.26	1.48	3.00	0.36	3.28	-0.64
2020-11-01	1.00	-0.80	3.27	-1.47	4.00	0.82	3.31	-0.13
2020-12-01	7.00	0.51	3.25	3.24	7.00	1.96	3.30	1.74
2021-01-01	1.00	-0.10	3.23	-2.13	0.00	0.00	0.00	0.00
2021-02-01	7.00	0.41	3.19	3.39	4.00	0.80	3.28	-0.08
2021-03-01	1.00	-1.20	3.16	-0.96	0.00	0.00	0.00	0.00
2021-04-01	6.00	1.72	3.13	1.14	5.00	-0.23	3.24	1.98
2021-05-01	2.00	-0.85	3.10	-0.25	2.00	-1.31	3.21	0.10
2021-07-01	1.00	0.99	3.04	-3.03	0.00	0.00	0.00	0.00
2021-08-01	1.00	-1.09	2.97	-0.88	1.00	-2.18	3.14	0.04
2021-09-01	4.00	0.22	2.89	0.89	2.00	-0.56	3.04	-0.48
2021-10-01	5.00	0.59	2.84	1.56	4.00	2.04	2.94	-0.98
2021-11-01	1.00	-0.06	2.83	-1.77	1.00	-1.63	2.82	-0.20
2021-12-01	4.00	-0.80	2.85	1.95	2.00	-0.83	2.71	0.12
2022-01-01	1.00	0.21	2.90	-2.12	3.00	-0.21	2.59	0.62
2022-02-01	3.00	-1.02	2.98	1.04	4.00	0.70	2.47	0.83
2022-03-01	2.00	0.73	3.05	-1.78	4.00	1.33	2.35	0.32
2022-04-01	2.00	-1.70	3.11	0.59	4.00	2.30	2.23	-0.53
2022-05-01	6.00	3.24	3.17	-0.41	4.00	1.59	2.11	0.30
2022-06-01	1.00	-1.97	3.23	-0.26	1.00	-0.11	1.98	-0.88

Continued on next page

Table B10: Time series data of monthly joining and leaving members for Redis

date	Joining	Seasonal	Trend	Residual	Leaving	Seasonal	Trend	Residual
2022-07-01	7.00	2.00	3.29	1.71	0.00	0.00	0.00	0.00
2022-08-01	3.00	-0.92	3.35	0.57	0.00	0.00	0.00	0.00
2022-09-01	3.00	-0.09	3.41	-0.33	0.00	0.00	0.00	0.00
2022-10-01	4.00	1.17	3.48	-0.65	0.00	0.00	0.00	0.00
2022-11-01	4.00	-0.38	3.54	0.84	0.00	0.00	0.00	0.00
2022-12-01	2.00	-0.73	3.60	-0.87	0.00	0.00	0.00	0.00

Table B11: Time series data of quarterly onion roles for Redis

quarter	Total Members	Core	Share	Regular	Share	Casual	Share
2016Q2	14	0	0.00	2	0.14	12	0.86
2016Q3	18	2	0.11	6	0.33	10	0.56
2016Q4	30	10	0.33	6	0.20	14	0.47
2017Q1	60	28	0.47	12	0.20	20	0.33
2017Q2	84	33	0.39	23	0.27	28	0.33
2017Q3	124	54	0.44	38	0.31	32	0.26
2017Q4	104	48	0.46	28	0.27	28	0.27
2018Q1	104	56	0.54	22	0.21	26	0.25
2018Q2	144	71	0.49	38	0.26	35	0.24
2018Q3	134	53	0.40	40	0.30	41	0.31
2018Q4	190	89	0.47	52	0.27	49	0.26
2019Q1	178	85	0.48	45	0.25	48	0.27
2019Q2	212	108	0.51	53	0.25	51	0.24
2019Q3	212	96	0.45	57	0.27	59	0.28
2019Q4	214	99	0.46	58	0.27	57	0.27
2020Q1	206	99	0.48	52	0.25	55	0.27
2020Q2	262	124	0.47	64	0.24	74	0.28
2020Q3	234	95	0.41	71	0.30	68	0.29
2020Q4	302	131	0.43	99	0.33	72	0.24
2021Q1	260	124	0.48	70	0.27	66	0.25
2021Q2	266	135	0.51	66	0.25	65	0.24
2021Q3	216	102	0.47	56	0.26	58	0.27
2021Q4	280	136	0.49	72	0.26	72	0.26

Continued on next page

Table B11: Time series data of quarterly onion roles for Redis

quarter	Total Members	Core	Share	Regular	Share	Casual	Share
2022Q1	308	153	0.50	83	0.27	72	0.23
2022Q2	314	144	0.46	90	0.29	80	0.25
2022Q3	356	171	0.48	94	0.26	91	0.26
2022Q4	344	160	0.47	93	0.27	91	0.26

Table B12: Time series data of monthly organizational knowledge concentration for Redis

date	Total Contributions	Contributions by Redis	Share
2016-04-01	43	0	0.00
2016-05-01	86	4	0.05
2016-06-01	39	0	0.00
2016-07-01	27	0	0.00
2016-08-01	28	0	0.00
2016-09-01	5	0	0.00
2016-10-01	18	0	0.00
2016-11-01	19	0	0.00
2016-12-01	164	0	0.00
2017-01-01	158	0	0.00
2017-02-01	89	0	0.00
2017-03-01	140	10	0.07
2017-04-01	85	2	0.02
2017-05-01	169	65	0.38
2017-06-01	276	28	0.10
2017-07-01	259	1	0.00
2017-08-01	244	2	0.01
2017-09-01	126	0	0.00
2017-10-01	181	0	0.00
2017-11-01	239	0	0.00
2017-12-01	175	0	0.00
2018-01-01	274	0	0.00
2018-02-01	128	2	0.02
2018-03-01	108	2	0.02
2018-04-01	143	0	0.00

Continued on next page

Table B12: Time series data of monthly organizational knowledge concentration for Redis

date	Total Contributions	Contributions by Redis	Share
2018-05-01	174	45	0.26
2018-06-01	154	23	0.15
2018-07-01	287	9	0.03
2018-08-01	229	13	0.06
2018-09-01	112	0	0.00
2018-10-01	163	10	0.06
2018-11-01	238	7	0.03
2018-12-01	174	7	0.04
2019-01-01	145	1	0.01
2019-02-01	269	3	0.01
2019-03-01	150	2	0.01
2019-04-01	155	7	0.05
2019-05-01	199	4	0.02
2019-06-01	195	25	0.13
2019-07-01	306	22	0.07
2019-08-01	309	45	0.15
2019-09-01	259	37	0.14
2019-10-01	277	36	0.13
2019-11-01	188	5	0.03
2019-12-01	263	7	0.03
2020-01-01	188	21	0.11
2020-02-01	291	39	0.13
2020-03-01	256	43	0.17
2020-04-01	330	11	0.03
2020-05-01	159	10	0.06
2020-06-01	257	37	0.14
2020-07-01	410	48	0.12
2020-08-01	243	9	0.04
2020-09-01	268	11	0.04
2020-10-01	232	3	0.01
2020-11-01	165	9	0.05
2020-12-01	293	2	0.01
2021-01-01	193	7	0.04
2021-02-01	310	16	0.05
2021-03-01	282	54	0.19

Continued on next page

Table B12: Time series data of monthly organizational knowledge concentration for Redis

date	Total Contributions	Contributions by Redis	Share
2021-04-01	233	14	0.06
2021-05-01	386	59	0.15
2021-06-01	350	62	0.18
2021-07-01	380	74	0.19
2021-08-01	395	52	0.13
2021-09-01	286	88	0.31
2021-10-01	389	65	0.17
2021-11-01	408	64	0.16
2021-12-01	339	77	0.23
2022-01-01	323	49	0.15
2022-02-01	373	51	0.14
2022-03-01	463	54	0.12
2022-04-01	322	48	0.15
2022-05-01	320	67	0.21
2022-06-01	357	73	0.20
2022-07-01	415	90	0.22
2022-08-01	518	93	0.18
2022-09-01	400	65	0.16
2022-10-01	377	56	0.15
2022-11-01	487	77	0.16
2022-12-01	535	69	0.13