The place of the open source ecosystem in the innovation landscape

dissertation submitted by

Mirko Boehm

to Faculty VII Economics and Management

of Technische Universität Berlin

in March 2024

for the degree of

Doctor of Economic Sciences

Abstract

When open source solutions began to compete successfully against established commercial products in the early 1990s, the pervasive and disruptive changes to the ICT sector which they introduce quickly became clear. Since then, open source has become mainstream and widely adopted, while remaining a driver of innovation. It has become the default way of developing software for many use cases, including the cloud and the web. However, academic insight in the new field of how open source fosters innovation was still rare. Many aspects of open source were expressed in community culture, but not yet in published research.

To bridge this gap requires a combination of the two perspectives applied in this research, an inside view of open source communities and that of a researcher of innovation economics. Beyond the effects that open source has on software technology and development methodology, its impact on innovation is especially visible in three key areas, namely the governance of collaboration in social groups, standards development and intellectual property regimes. These three areas have been individually researched in scientific articles and in EU funded research projects, with the published results presented here as the main chapters. Additionally, these separate perspectives are synthesized into a theoretical framework for the economics of open source, creating connections with concepts that are unique to it, like joint stewardship or the relationship between distributing software freely and expectations of fairness and reciprocity.

These results build on two larger studies, one on “The relationship between open source software and standard setting” and one on “The impact of open source software and hardware on technological independence, competitiveness and innovation in the EU economy” that were conducted in parallel, built on the concepts presented here and fed new insights back into this research.

The ICT sector is becoming increasingly regulated. Especially the EU is establishing a comprehensive regulatory framework for digital products, services and markets. Since all ICT regulation will affect and interact with the open source ecosystem, understanding the dynamics between businesses and open source communities as they are shaped by intellectual property rights, standards and governance frameworks gains further

increased relevance to the functioning of the market and to achieving regulatory goals.

In a time of deteriorating international security and increasing trade tensions, the global nature of open source collaboration limits the ability of regulators to control the flow of technologies, but also offers a way to collaborate across regional and political divides based on the principles of open governance and openly licensed outcomes.

This research aims to illustrate why and how open source drives innovation. With that, it hopes to suggest a theoretical foundation for how open source collaboration can be help to bridge regional and cultural divides and foster diversity, equity and inclusion.

License

Publication record

The paper "The emergence of governance norms in volunteer driven open source communities" was first presented on 11 January 2017 at the TU Berlin Research Colloquium (CIF). It was published as: Mirko Boehm. "The emergence of governance norms in volunteer driven open source communities". In: *Journal of Open Law, Technology & Society* 11.1 (2019), pp. 3–39 (DOI `https://doi.org/10.5033/jolts.v11i1.131`, accepted manuscript included as chapter 2), available under the terms of a Creative Commons Attribution 4.0 CC-BY license. 🅭🅯

The paper "Open Source Software in Standard Setting: The Role of Intellectual Property Right Regimes" contains a brief summary of the wider study on *The Relationship Between Open Source Software and Standard Setting* [15]. It was published as a chapter in *Open Source Law, Policy and Practice* [26] as: Knut Blind, Mirko Böhm, and Nikolaus Thumm. "Open Source Software in Standard Setting: The Role of Intellectual Property Right Regimes". In: *Open Source Law, Policy and Practice*. Oxford University Press, Oct. 2022 (DOI `https://doi.org/10.1093/oso/9780198862345.003.0011`, accepted manuscript included as chapter 3), available under the terms of a Creative Commons Attribution-Non Commercial-No Derivatives 4.0 license. 🅭🅯🄏⊜

The paper "Standard setting organizations and open source communities: Partners or competitors?" was shared with the Legal Network for review and discussion in May 2018. It was published as: Mirko Boehm and Davis Eisape. "Standard setting organizations and open source communities: Partners or competitors?" In: *First Monday* 26.7 (June 2021) (DOI `https://doi.org/10.5210/fm.v26i7.10806`, accepted manuscript included as chapter 4), included here under the terms of a Creative Commons Attribution 4.0 CC-BY license.

The paper "Economics of Open Source" was presented in an early draft stage at CIF on 6 November 2019. It was published as a chapter in *Open Source Law, Policy and Practice* [26] as: Mirko Boehm. "Economics of Open Source". In: *Open Source Law, Policy and Practice.* Oxford University Press, Oct. 2022 (DOI `https://doi.org/10.1093/oso/9780198862345.003.0015`, accepted manuscript included as chapter 5), available under the terms of a Creative Commons Attribution-Non Commercial-No Derivatives 4.0 license.

Acknowledgements and thank you notes

Alexandra, my one and only, my Bubbles, my rock, center of my universe, my partner, the mother of my babies, for whom you bear the scars: Only you give sense to this writing. Without your love, patience, encouragement, grace and occasional subtle (?) pressure, I would never have finished this. Sun lights up the daytime, moon lights up the night. I light up when you call my name. Three months after we met, I told you I pictured us, gray of hair, holding hands on a park bench. I am sure we will and I look forward to it. I love you.

My children, my two reasons to strive, hesitant and blinded by the light of your own brilliance: Jay, righteous to the core and pure of heart. Paul, radiating creativity while finding your path. I may have given you more freedom to be yourselves than you are comfortable with at times. Maybe I could have given you better guidance. I continue to believe liberty of thought is the strongest foundation I can provide for you. You make me endlessly proud. One of these mornings, you will rise up singing. You will spread your wings and you will take to the sky. I will be there to buoy you up. I love you.

My parents, who saw everything they built get washed away in 1990; Who got back up, dusted off their coats and started over, putting responsibility for our family first and rebuilding everything in their lives. You found the immense strength to start all over again, and still first and foremost had the best for your children in mind. You put down your load where you thought it was best. Made a home in the wilderness. Built a cabin and a winter store. Ploughed the ground by the cold lake shore. You provided the shoulders on which I stand. I love you.

My friends, who encouraged me to keep going, to aim a little bit higher, who proof-read the drafts, had my back and never questioned my questionable choices. Paul and Helen Finlayson Adams, Karl and Jennifer Beecher, Amanda Brock, Thomas Fricke and Erdmute Nieke, Martina and Stephan Hertel, Heike Koblitz und Klaus Büchel, Frank Karlitschek, Lucy and Andrew Katz, Lisa Nöth and Stephan Lindauer, Florian and Alexander Willner, Arne Lösekann, Bob McDowell and Gudrun Briegel, my brother Sebastian and Sabrina Sucker, Sven Tresp, Cordi and Karsten Villwock, Anke und Torsten Zugehör. You reminded me that there are finer things

in life (usually slow-roasted or well-aged), and that knowledge is one of them. You taught me to cut the fickle people from my life, the rambler, the gambler, the back biter. Those who remain a part of it are the ones who try their hardest to be the best they can be. They will be there for you. You don't judge, you help. You are the companions I want in my life. Es gibt kein richtiges Leben im falschen. I look forward to celebrating the right one with you.

My mentor, Keith Bergelt, a man of wealth and taste: From the moment we met, you believed in me and provided intellectual honesty, sage leadership, selfless advise; you demanded performance and provided the space to grow that I needed to develop as a person, an expert and a leader. Your influence has been instrumental in my personal and career growth. Your encouragement, wisdom, and humor have made an invaluable difference to my life.

My collaborators, Simon Phipps, Dave Marr, Davis Eisape, the Open-Forum Europe team and the amazing contributors to the Legal Network and the KDE Community: You have shown me again and again that substance matters and that diversity of thought and form is the secret sauce to achieving true change. You embody the spirit of open collaboration that this writing is about.

My doctoral advisor, Knut Blind, who gave me the chance to re-enter academic life at an unusual career stage, believed in me sufficiently to to teach at his research chair at TU Berlin, collaborated with me on two major research projects and guided this research work with academic rigour and incredible patience. You see your success in the academic careers of the researchers you take under your wing and lift up with your own brilliance as a shining example of Forschung und Lehre.

Und am Ende der Strasse steht ein Haus am See. Here is to life-long learning. I am glad I made the decision at the age of 39 to go back to school. Researching, writing and teaching a gaggle of students who combine being incredibly smart with curiosity and a lifetime to consider, analyse and reflect is literally the fountain of youth. What mid-life crisis!?!

Contents

List of Figures

Chapter 1

Introduction

1.1 How does FOSS drive innovation?

The goal of this research is to understand in more detail how free and open source software (FOSS)[1] drives innovation. The observable changes clearly indicate that it does, to the extend that businesses and policy makers have been caught by surprise when FOSS solutions began to displace incumbent commercial products and when FOSS collaboration methods changed expectations about governance in society.

This transition affects all but the primary sector of the economy. It includes the production of goods, the supply of services, the management of knowledge as well as the activities of governments and volunteer groups. In a virtuous cycle of developing new methods of collaboration, applying them to new applications of networked productive activities and setting new expectations of diversity, equity and inclusion (which triggers the next round of developing new forms of collaboration), pervasive changes are introduced repeatedly. Economic actors change their behavior in a wide range of human activities.[38] For the first time in history, knowledge transfer in education, research and technical development is

[1]The terms free and open source software (FOSS), open source, free software (FS) or sometimes free/libre and open source software (FLOSS) are used interchangeably in this document. While the terms have separate histories, they are refer to software that is distributed under a license that gives all the rights required by the Open Source Definition. Wherever possible, the term free and open source software (FOSS) is preferred. The individual articles have been kept as close as possible to the published versions, with only minor stylistic changes. Based on the publisher guidelines, they partially use slightly different terminology. For example, chapter 5 uses “Open Source” in upper case.

anchored in a public and freely distributed body of knowledge and managed using means of production that are public goods. The availability of FOSS implementations of royalty-free (RF) standards on formatting, storing, maintenance and retrieval of information give hope that future gaps in the preservation of knowledge can be prevented.[39]

One of the first indications that FOSS changed market dynamics was the challenge that the introduction of free software competitors posed to the positions of entrenched incumbents. Market segments where volunteer communities had both the required expertise and the need to improve quality and choice where affected early on: The examples of programming language compiler and interpreter toolkits (gcc, Perl, Python), desktop environments (KDE, Gnome), integrated software development environments (Eclipse) or server systems to host applications and web services (Linux, Apache) proved that even before businesses embraced FOSS, collaborative development was able to compete with established proprietary offerings.[116] However, their situation was unique in that these trailblazer communities introduced the FOSS development model into an environment firmly dominated by proprietary, commercially licensed software.

In the next phase, FOSS collaboration gradually became the default mode of developing new software systems. Whenever a new paradigm of software application emerged, a free software solution was usually competing early on. In the data center, frameworks for operating virtual machines at scale (OpenStack) and later container clusters (Linux containers, Kubernetes) and generally Linux instances as the host environment for cloud workloads established FOSS as first choice. This shift was even more dramatic in the field of web development frameworks (PHP, Ruby on Rails, Node).[129] In this stage of FOSS adoption, no proprietary system was able to conquer a complete market again in the way Microsoft Windows dominated the desktop computer. Also, businesses recognized the benefits of embracing FOSS methodologies and products.[87] Community composition changed from predominantly volunteer-based to hybrid, where corporate contributors and volunteer contributors collaborate. Hybrid communities with a smaller but not irrelevant share of volunteer contributors remain the norm as of today.[123]

More recently, FOSS adoption reached another level when, also related to trade tensions and a deteriorating international security situation, digital sovereignty came to be recognized as a policy goal. As FOSS is now pervasively adopted in the operations of digital infrastructure, the reliable maintenance, security and integrity of open source software along the supply chain becomes increasingly important also to governments.[41] At this current evolution of the open source ecosystem, FOSS foundations have become the stewards of governance that interface with public and private stakeholders in the further development of open source software. Digital products in the widest sense are expected to become a more regulated industry. In a novel legislative approach, the European Union (EU) Cyber Resilience Act (CRA) recognizes the dedicated role of open-source software stewards as economic operators in their own right, separate from manufacturers and distributors.

The current level of maturity of the FOSS ecosystem underlines the fundamental role of FOSS as a mode of production. This thinking is anchored in the understanding of FOSS as an application of peer production distinct from markets or firms (see 1.2).[11] While FOSS is predominantly created by private actors, the produced result is a public good as required by the terms of recognized open source licenses. FOSS licensing in itself makes private citizens, civil society, private enterprise and government all stakeholders in the software development process. Nevertheless, licensing and governance are separate and disjunct dimensions, with many of the stakeholder expectations like transparency, accessibility and inclusion referring to governance attributes of the production process.[79] Thinking of FOSS as a mode of production highlights the place of FOSS in economic theory. FOSS development and distribution represents the aspects of production and trade along a supply chain. FOSS licensing shapes the framework of supply chain transactions and the relationships between the individual producers and consumers.[81]

The separation of governance and licensing raises the question of their relationship in the context of FOSS. Do FOSS licenses create an environment in which collaborative development flourishes? Or do the underlying economic principles of supply and demand create an environment in which collaboratively developed software components are best shared under free software licenses? Resolving this question will have implications on research and innovation policy as well as regarding the role of governance norms when defining the openness of the development process. Of particular importance is how the impact of FOSS changes in situations where openly licensed software is developed in a closed process or if all intellectual property rights (IPR) are concentrated in the hands of a single contributing business.[122] There are specific risks involved when, as with Android, a critical piece of software infrastructure is developed by a single business (Google) in a closed process, or when, as with Terraform, a single business (Hashicorp) is able to switch to a business license after consumers began to depend on their product.[53] As the market for digital products becomes increasingly regulated, policy makers are looking beyond just licensing when defining whether open source development is exempt from market regulation (as with the EU CRA). This relates the question to that of pricing digital goods as well as to the effect of FOSS licensing and governance on competition.

One market segment, that of telecommunications technologies, took a path very different from others as FOSS became more and more widely used. Mobile communications protocols and software stacks for the most part remain proprietary and covered by thickets of standards essential patents (SEPs).[46] Communications technologies embedded into consumer devices like Bluetooth or wireless internet (Wi-Fi) continue to be covered by commercially licensed patent portfolios, even though FOSS implementations exist. The underlying strong relationship between FOSS and the licensing of IPR is able to encourage or retard the diffusion of FOSS in a specific market segment. This includes the issue of compatibility between FOSS and IPR licensing models when combined into the same product, and whether the introduction of IPR licensing in technical standards on the form of SEP is conducive to their adoption in digital products that build on FOSS technologies.[48]

While numerous FOSS technologies have seen explosive adoption and developed into market or industry standards, it was still unclear at the beginning of this work how standards and FOSS development mutually influence each other.[90] Since FOSS implementations become available immediately during development, otherwise well-established concepts like pre-competitive cooperation or the development of technical standards in fora and consortia are difficult to apply. The relationship between standards and FOSS development has become one of the themes of this research.

Other aspects touched upon in the following chapters include whether or not FOSS provides a pro-competitive influence in markets, and how businesses can benefit from participating in FOSS. It can be observed that businesses engage either directly by contributing parts of their work or indirectly by supporting communities over extended periods of time, indicating that they expect a positive impact on marketed products and services from their contributions that goes beyond market signaling, public relations and talent acquisition.[124] Many businesses engage and thrive in setups where no direct revenue is generated from their participation. The combined forces of, on the one hand, market pressure that makes FOSS adoption a business imperative once competitors in the same market segment do so, and on the other hand, of efficiency gains from reducing necessary research and development (R&D) investments into non-differentiating technologies together form a business rationale for sustained engagement in the FOSS community.

A final perspective inherent to the role of FOSS in ICT innovation is the relationship between FOSS and government, especially regarding to what extend and how it can effectively be regulated. On the one hand, FOSS has become an integral part of the ICT sector, and as such will be subject to market regulation related to digital products. On the other hand, FOSS collaboration is inherently global, inclusive and mostly apolitical. It flows across borders and regions with little regard for nation states. Also, the FOSS development process is much harder to regulate than that of proprietary technologies.[85] This limits the influence policy makers can exert over the development process, over what technology and information are shared across regions, as well as over FOSS governance norms. To shape the cross-border flow of FOSS possibly requires multi-stakeholder coalitions, similar to aspects of international trade.[157]

In a mutual relationship, the FOSS ecosystem shapes and is shaped by the overall innovation landscape, understood here as the regulatory frameworks, business models, institutions and organizations working to bring innovations to market.[28] This problem space leads to a progression of research questions that are covered in the following chapters.

The initial focus is on how FOSS production is organized or governed, with a focus on the volunteer-driven communities that shaped the initial phase of FOSS adoption. Chapter 2 introduces the concept of community composition and its implications on governance frameworks and norms, as well as the interaction of makers and community builders in the collaboration process. Based on multiple case studies, it highlights the difficulties in establishing accountability and ensuring long-term sustainability in self-governing communities. The focus then shifts to investigate how the wider open source community relates to and interacts with the traditional drivers of innovation, primarily standards development organizations (SDOs), academic knowledge transfer and IPR regimes. Chapter 3 explores the influence of IPR regimes on the role of FOSS in standard setting. It shows that compatibility between FOSS and fair, reasonable and non-discriminatory (FRAND) licensing conditions is necessary but not sufficient to establish collaboration and delves into the different understandings of governance in both fields. Chapter 4 asks from a birds-eye view whether standards setting organizations (SSOs) and FOSS communities are partners or competitors. It introduces a phase model of standardization that is able to explain the standardizing effects of SSO and FOSS in a single theoretical framework by taking a utilitarian view on standardization. It identifies cost of change as a key determinant of the choice of early versus late standardization. Finally, chapter 5 synthesizes these separate perspectives into a theoretical framework for the economics of FOSS. It positions the wider open source community in law, politics and economics, illustrates the rationale for businesses to engage in FOSS development based on how they are able to realize benefits from that engagement, and introduces joint stewardship of the currently active contributors as a pillar of governance. Referencing back to the different dimensions of analysis visited before, it is able to define license compliance as a hygiene factor, the strict requirement of ex-ante agreements and the absence of negotiation as an implication of software freedom, and the outsize impact of a small number of bad actors on the functioning of the fabric of the FOSS ecosystem as related to the special role of a price of zero in economic theory. Chapter 6 summarizes the contributions of the papers to the state of research of the economics of FOSS, including an overview of their relation to current policy debates, discusses reach and limitations of the provided analysis and points out opportunities for further study.

1.2 Scope of research

Combining the experiences of a practitioner with a long history of FOSS contributions and that of a a researcher, this work started in 2013 with the assumption that the state of research at the time did not fully describe the dynamics of the FOSS ecosystem. Partially, that was caused by the explosive growth of FOSS adoption, with made it difficult for research to catch up with developments. Researchers have been able to observe symptoms, however it requires an insider perspective to identify some of the underlying larger trends. At this time, no holistic theoretical approach that satisfied the interdisciplinary nature of the FOSS phenomenon had found wider acceptance. Theoretical starting points for the analysis of the relationship of FOSS and innovation existed with Benkler's concept of peer production[11] and with Weber's analysis of FOSS as "an experiment in building a political economy"[154]. Lerner and Tirole described how "the behavior of individual programmers and commercial companies engaged in open source projects is initially startling"[84] to economists and illustrated how contributors furthered their careers by signaling their expertise through their FOSS engagement. They also argued that "open source activities can be understood within existing economic frameworks"[83]. Approaches to applying and possibly augmenting existing economic frameworks to explain FOSS have been shaped by Shapiro and Varian's work on the network economy and on pricing information goods.[133, 150, 149]. At the same time, authors with a FOSS background wrote about their philosophy and experiences, however mostly without a connection to economics. Stallman shaped the thinking about software freedom.[140] By arguing that cyberspace can and should be regulated, and that eventually it will, Lessig contributed to the developing political and regulatory dimension of FOSS.[85] Raymond and Steele described the drastically different software development methodologies applied in FOSS collaboration, including concepts like release-early-release-often, some of which are still commonly referenced today.[119] By codifying and elaborating the Open Source Definition, Perens set the norms for the interpretation of FOSS licenses and established the connection between the Open Source Definition and software freedom.[113] At a bird's eye view, while FOSS was a popular research topic, the literature available at the time represented disconnected theoretical approaches and used diverging terminologies. A common body of theory had not yet emerged. A breakthrough in the systematic and structured analysis of open innovation which established an explicit relationship to open source development was initiated by Chesbrough, Vanhaverbeke, and West.[29]

Grounding the theory of FOSS within existing economic frameworks creates connections to a number of classic references. Benkler directly builds upon Coase's "The Nature of the firm" to theoretically establish FOSS as a fourth transactional model applied to the product of information goods.[32] In this research, Schumpeter's theory of creative destruction is connected to the market disruption caused by the competition of FOSS.[130] Hirschman's voice-or-exit theory is applied to explain the engagement of contributors in community governance and the strong loyalty they develop to communities over time.[65] Herzberg's idea of hygiene factors is use to explain the open source community's principled stance on license compliance and ex-ante agreements.[64] The concepts on how governance emerges in social groups which have been developed by Ullmann-Margalit are applied to FOSS communities,[146] as is the theory of collective action and the evolution of social norms developed by Ostrom.[110]

In a field experiencing such rapid change, the voice of practitioners gains particular relevance as first-hand accounts of important developments. A key role in developing the thought on the legal and compliance implications of FOSS adoption in business was played by the Legal Network, an expert group hosted by Free Software Foundation Europe (FSFE) that published the International Free and Open Source Software Law Review journal. Kemp researched the implementation of sensible, proportionate FOSS governance in organizations.[77] Johnny, Miller, and Webbink analyzed the boundaries and limitations of using copyright to manage rights to FOSS contributions and project code bases.[73] Mitchell and Mason wrote about the compatibility of FOSS licensing terms with those of embedded patents.[97] Hunter and Walli analyzed the evolving role of open source software foundations as neutral non-profit platforms for open technology collaboration.[69]

In book form, Bacon collected and popularized best practices on how to lead and grow FOSS communities.[5] Whitehurst described how business organization can be inspired by open source philosophy.[155] Laffan developed approaches to measure openness in governance.[79]

The interdisciplinary character of FOSS means that research needs to cover different fields of study, from various disciplines of economics to technology and computer science. Some research is relevant to the study of FOSS without being concerned with it directly. For example, Dixit and Olson investigate the how voluntary participation affects the Coase Theorem in 2000 without mentioning FOSS.[40] Heller's tragedy of the anticommons is applied in this research to the enabling function of automatic and transitive FOSS licensing, even though it does not directly reference FOSS.[62] This again shows the application of existing economic frameworks to the FOSS ecosystem and illustrates the pervasive impact of FOSS on the functioning of markets.

More recently, newer research contributed to a deeper understanding of FOSS. Shampanier, Mazar, and Ariely et al showed how a price of zero frames social as opposed to market exchanges and invokes a different set of social norms, which this research relates to the hygiene factors of license compliance and ex-ante agreements.[132] Eghbal placed FOSS at the foundation of modern digital infrastructure, highlighting the public interest in the viability of the FOSS ecosystem.[41] Rosen formulated principles that open standards should implement.[127] Lundell, Gamalielsson, and Katz researched to what extend International Organization for Standardization (ISO) standards can be implemented in FOSS.[91] Kappos and Harrington argue that FOSS and FRAND represent compatible IPR regimes in standards setting,[74] while Maracke only partially supports that argument.[94] Nagle identifies a positive contribution of the use of and contribution to FOSS on firm productivity.[99] Finally, Tirole anchored FOSS in the economics for the common good, also highlighting the governance and social responsibility of businesses in that context.[144]

Current developments in the FOSS ecosystem continue to be documented by community sources. For example, the practices of the Linux kernel community in enforcing license compliance are part of the kernel documentation.[78] The FSF documents the principles of community-oriented GNU General Public License (GPL) enforcement.[58] Hemel and Coughlan provide additional guidance for businesses in ensuring license compliance.[63] In an effort to substantiate insights into the FOSS ecosystem, Linux Foundation Research investigates the state of play and developments in the wider open source community, focusing on topics like diversity, equity and inclusion,[27] the state of open source adoption in the European public sector,[109] survey results on open source trends, sustainability challenges, and growth opportunities[81] and other areas. Of particular relevance is the general census of the use of FOSS developed in collaboration with Harvard Business School.[100]

In conclusion, the literature on the intersection between FOSS and innovation economics did not provide a coherent, widely applicable body of theory. Also, the disconnected analyses may represent the observations of different researchers of the changes FOSS brought to their field of study, while the intention here is to place FOSS at the center of the analysis as a framework for the creation and exchange of freely licensed information goods. The choice of research methods reflects this starting point. The articles included here primarily apply qualitative methods to support theory building so that the gap between the separate theoretical approaches can be bridged.

1.3 Framework conditions

This study is set in framework conditions at the level of the ICT industry sector, economies in major trade blocks, foreign and international relations, as well as the wider open source community and civil society.

Major trends accompany the digital transition of the ICT industry sector, namely digitalization, the development of improved methods of collaboration, a major trend towards openness and transparency and a shift of the role of the modern state from an employer and producer to a regulator.[22] On the one hand, FOSS presents an opportunity to realize the efficiency gains necessary to stay competitive. On the other hand, FOSS reduces the space of viable commercial models by providing a common set of baseline technologies. It also increases differentiation pressure on manufacturers by making more and more non-differentiating functionality available to anybody and for any purpose. Businesses and communities operate in a competitive ICT environment with an almost constant demand for change.

Economies especially in Europe, Japan, China, the rest of Asia and the USA face global competition. International trade regulation converges towards harmonization to facilitate the flow of goods, sometimes by exporting progressive regulation.[24] After the end of the cold war globalization and free trade doctrines dominated. From that peak of international security the world now trends towards a multilateral order of regions with more comparable share of global gross domestic product (GDP). Tariffs and other barriers on trade are being used in the economic conflict between the United States and China. Most worryingly, the security environment has deteriorated to the extend that prolonged international warfare is now conducted again on European soil. This deterioration of economic and international stability influences the questions researched in this study, however it has not yet lead to a major disruption. The presented theory assumes an economic framework based on the protection of private property and the rule of law. It is unclear if and how it could apply otherwise.

The trends prevalent in the ICT sector also affect the FOSS ecosystem and civil society. Closer international collaboration increases pressure for more diversity, equity and inclusion and respect for human rights. The United Nations (UN) Sustainable Development Goals (SDGs) have become a focal point for social impact assessments.[37] Initiatives to reduce the north-south divide and pressure for increasing gender equality are part of the environment the FOSS community operates in.

1.4 Adjacent fields of research

A related but separate field of research is the software engineering dimension of FOSS. As an enabling technology, FOSS elevates innovation into higher up layers of the technology stack by providing a wealth of foundational software components. It also continues to shape software engineering tools and methodology as well as software delivery. These effects are generally better understood and not researched here. Similarly, the application of open source methodologies to other fields than software source code is growing in importance, but not in scope here. While there are similarities in licensing concepts and ideology, software is unique in comparison in that it allows for the relatively simple and incremental creation of derivative works. The open source development cycle of taking an existing project, adding incremental changes in the form of patches to it and offering these changes to the original project for review and integration is based on it. Hardware designs or data may be openly licensed, however contributions can rarely be as granular and minimal as with software source code. They should be researched separately. The same is true for digital video and pictures. How collaborative and incremental development can be applied to artificial intelligence applications is not yet fully understood and also not in scope for this research. From a technical point of view, this analysis focuses on software source code and structured text representing knowledge, like technical documentation or encyclopedias, where contributions can be granular and incremental.

FOSS not only changes innovation economics, but also other fields of economics and of business management. HR departments use published FOSS contributions to select candidates for hiring, carefully manage their attractiveness as an employer by adopting to newer behavioral norms, and develop skills and in-house expertise with FOSS. In public relations, corporations strive to be seen as good open source citizens and are aware of the reputational risks from license compliance violations and other cultural misfits. IT operations deploys in-house solutions partially based on FOSS technologies. These consequences of the wide adoption of FOSS are not in the focus, however the theories developed here may provide insights about the underlying trends.

There are also implications of the growing relevance of FOSS on political science and policy making that are beyond the scope of this research. Both FOSS technology as well as collaboration methods enable new participatory forms of political representation. Applied in the right way, FOSS can be a driver for digital sovereignty and technological independence. The public sector may be able to reduce cost and increase supplier competition by strategically developing and procuring FOSS based digital public services. These aspects are not investigated here.

1.5 Summary

The following chapters cover three aspects of FOSS as a method of creating and sharing knowledge with important impacts on the innovation landscape, namely open governance (chapter 2), IPR regimes (chapter 3) and standards development (chapter 4), followed by an article that introduces an integrated theory of the economics of FOSS (chapter 5). Since their publication, these articles already have had noticeable impact on research, community practices, businesses and in policy making. They are complemented by the comprehensive study on "The impact of Open Source software and hardware on technological independence, competitiveness and innovation in the EU economy"[19] which references and builds upon some of the theories presented in the articles.

Chapter 2

The emergence of governance norms in volunteer-driven open source communities

Free and open source software communities develop their governance norms and practises as they grow from small to medium to large size social groups. Communities with a small number of participants typically organise informally. As the community grows, the need for coordination grows as well and at some point more pronounced organisation becomes necessary. The growth stages are defined by the coordination mechanisms applied – ad-hoc coordination for the initial small group, consensus focused auto-organisation for the medium size group, and structured, more formalised coordination for the large size group. The main interest of the communities is to attract and retain contributors and to facilitate contributions to their products. The communities studied in this qualitative embedded multiple-case study exhibit governance related debates and conflicts as they reached a large size, leading to difficulties in further growing the number of involved contributors and sustaining the community activities. The paper researches the emergence of governance norms in these communities and the role these norms, once established, play in the management of the communities in their current stage. The study finds that the governance norms in communities are commonly developed by participants that do not think them necessary for a community that does not want them at the time. The result is a pre-eminence of implicit, under-documented norms that increase barriers of entry for newcomers and afford incumbent contributors with instruments to derail unwanted decisions. The paper isolates the essential contradiction that the communities aim to maintain devolved authority at the contributor level, but require effective decision making and policing mechanisms to implement and maintain that. It recommends that communities, instead of deferring or down-playing the need to set up explicit governance norms, purposefully develop norms that explicitly define structure and processes so that they support, enforce and protect the devolved authority their participants should have.

2.1 An inside view on social norms in communities

On February 3, 2016, something happened in the KDE FOSS community that would become the dominant topic for more than half a year: An announcement[1] was sent to the community mailing list that a draft for a new vision for the community was being worked on. This announcement would trigger close to 350 postings to various mailing list threads, constituting almost half of all discussions within the community in the first half of 2016. It would lead to heated discussions between competing vision drafts, public endorsements and statements of support, virtual ad-hominem attacks and even contributors leaving the community in anger. How could an announcement of something so basic, so fundamental to a large decentralised group of volunteers like a vision create such distress?

In May 2016, a code of conduct for the FSFE was announced to the organisations' coordinators. After smaller changes, a version was approved and sent to the core team for a decision in June. After just a few people voiced an opposition on some of the wording, the process came to a halt. When it was picked up again in October the same year and circulated in an almost unchanged form to the same people who had seen it already in May-June, it spawned one of the fiercest debates in the history FSFE with more than 200 mails in just two weeks. Suddenly, people spoke up in opposition not only of the content of the code of conduct but about the very need for a code of conduct in general. Indeed, the code of conduct

[1] https://mail.kde.org/pipermail/kde-community/2016q1/002241.html (accessed 03/02/2024)

had, in the eyes of some of the participants, become a tool not to include contributors but to silence unwanted opinions. While the general consensus seem to have been in favour of adopting a code of conduct, the process came to a halt again, since no decision could be reached. The code of conduct was finally adopted without substantial changes in October 2017.

There are many instances of such soul-searching in FOSS communities as they reach maturity and a large number of contributors. It can be observed that these controversies focus on questions regarding how the communities internally manage their social process, questions of *community governance* as the totality of implicit and explicit behavioural norms, codes and processes that regulate the relationship between contributors and the community. While these are certainly not the only challenges the communities face as they grow, evolving community governance appears to be a particularly difficult problem to overcome. When interpreting the habits and practises of voluntary collaboration of contributors in FOSS communities as a cultural phenomenon, governance norms are seen as the inside view on this culture. They express the way the communities see themselves. Understanding this inside view held by their own contributors of how the communities are expected to operate is relevant not only regarding issues of community management, but also to outsiders as the basis on which to engage with them. The public, regulators, businesses and influencers of technical innovation like SDOs or the patent offices are well-advised to understand the cultural norms and practises of FOSS communities in order to establish successful collaboration with them.

This paper researches the governance norms that evolve in volunteer-driven FOSS communities as they grow from an initiative of a few contributors to large and often international organisations. Assuming that these norms are based on the aggregate of the individual convictions and expectations of those contributing to the community, the paper describes this inside view the communities have of themselves and especially the behaviour the actors engaged in it expect from their fellows, from the community as a whole and from outsiders - individuals, other organisations and the public.

2.2 Governance in communities with voluntary participation

For this study, the work of the wider open source community is primarily viewed as a social process to produce information as a public good. It is a knowledge-intensive process that has inputs in the form of labor (the work of the contributors) and capital (the funding required by the communities). The output are information goods, most prominently the software components that are freely distributed to the public. The application of free software licenses makes them non-excludable and non-rivalrous and therefore public goods. The production of a public information good is one key element that defines a FOSS community. The other key element is voluntary participation of the contributors in the community. The understanding of FOSS community applied in this paper is that of a social group of contributors that participate voluntarily in the production of public information goods.[66] The participants in these groups, the contributors (see section 2.2.2), collectively create the community's products and make them available to the general public by distributing them under a free software license. This study focuses on communities that consist predominantly of individual volunteer contributors.

Communities that grow beyond a very small group of contributors develop (sometimes unconsciously) functional specialisation between contributors, division of labor between formally or informally defined subgroups, and integration of the individual contributions into an overall product. They become organisations. Functions that contributors specialise in can be product related (software development or content creation in general, maintainership over submodules, or release management) or supportive (marketing and public relations, finance, event management). To successfully release products over time, communities need to coordinate the work of the individual contributors so that, through a repetitive process of content creation, filtering for quality, integration and distribution, the product improves over time.[11] Coordination in this context is understood as a process, not as a task performed by a manager. Regarding the production process, the *need for community governance* results from the necessity to coordinate the work of a diverse group of volunteers to create the community product.

From the outside perspective of users or the general public, the communities are mainly known for the products they create. Potential contributors consider to engage with the community based on the product related participation opportunities, and on what is generally known about the culture of the community. A common recommendation is to "treat every user as a potential volunteer"[51] Most contributors participate in a community for a limited period of time, leading to fluctuation. To grow the number of incoming contributions, communities need to attract new contributors, and retain the existing ones so that the difference between influx and outflow remains positive. With regard to the interaction with the outside world, the *need for community governance* results from the necessity to maintain and grow the contributor base that forms the community.

Even though FOSS communities commonly operate as decentralised self-organised groups, they develop elaborate informal and formal rules and practises for their social process. These rules and practises are referred to as the *governance norms* of the community. A social norm is "a prescribed guide for conduct or action which is generally complied with by the members of a society"[146]. The term governance "refers to all processes of social organisation and social coordination"[12] in groups. It describes the processes of governing a formal or informal organisation performed by a formal government, a market or a network. Governance is expressed through a wide range of instruments ranging from laws to social norms, as well as language and culture. Any social group that coordinates working together towards a common goal will exhibit some form of governance. Whereas *government* refers to the institutions that exert power and influence over a constituency, governance can exist without institutions. Communities often hesitate to develop formal governing institutions. This directs the focus of analysis on governance (a process) over government (the institutions) when studying FOSS communities.

The subject of governance is reduced to decision making and conflict resolution within the social group by using broad definitions for both terms. "Decisions" is used here in the sense that whenever a small subset of the contributors or the whole community jointly agree on a course of action on some subject, a decision is made. Similarly, "conflict" is understood broadly as any disagreement of one or a group of contributors with any decision made by another subgroup or the community as a whole. Decisions do not need to be made in a formalised process, nor does conflict require a formal complaint or a heated argument. Decision making and conflict resolution are essential elements of collective action.[61] How the organisation defines who may participate in what decisions as a community member and what organs form the organisational structure characterises key aspects of governance. It can be expected that organisations exist to further the common interests of their members.[93] The reason for FOSS communities to exist is to facilitate the interests and motivations of their contributors. To illustrate the governance of FOSS organisations, this study will review the reasons why the organisations exist, the organisational structure of the community, the processes by which decisions are made and challenged, and how and with which roles contributors participate in them.

The ethics and convictions of the individual contributors should be reflected in the organisation's vision and mission statements. The formal and informal organisational structure provides the framework for the community's production process. Constitutional documents like bylaws and manifestos establish formal structure. Representative bodies like boards, committees and working groups are the most visible formalised form of it. Formal structure projects authority by assigning decision making power to individuals or organisational units. Besides those, informal structure likely exists that is more difficult to identify. Informal structure manifests itself in decisions that bypass hierarchy, or in strong impact of the opinions of individuals that are not appointed to representative positions. FOSS communities commonly show a preference for minimal formal organisation (see section 2.2.4), which leads to the assumption that informal structure has a more preeminent effect than usual. Formal organisation is also more difficult to change, since it typically requires both a qualified majority of the group members and a conscious effort to understand and reconsider the current structure and identify how it should be changed.

One potential reason for a perceived need for organisational change is a divergence between the formal and informal structure. Opposition to reform indicates that group members may be more comfortable with the existing balance of formal and informal structure. Decision making processes and conflict resolution mechanisms define how decisions are initiated and then made, and how to appeal against or escalate them in cases of disagreement, how decisions will be implemented or enforced, and how the community deals with minority opinions and opposition, especially in the case of controversial decisions. Decision making processes relate to organisational structure in that commonly, paths of escalation or appeal follow the hierarchy of formal organisation. Directly related to decision making processes is conflict resolution, as the cause of a conflict is either the wish for a decision to be made, or to appeal against one that was made. The balance between decision making processes, instruments of appeal and conflict resolution is what enables contributors to influence the community production process.

The social order within the community defines which stakeholders can take part in what group decisions. Differentiation can for example be based on eligibility or group status. A regular contributor may not be eligible to take part in a board decision, or may not have the status to take over a maintainer role. The question of social order in communities boils down to what decisions a contributor can participate in, and what the impact of this individual vote is. It relates to the definition of group membership that separates insiders from outsiders, but also possibly to status groups within the community. It is also related to how contributions are valued and translate into merit for the contributor. There may be a sense of equality, or a sense of elitism where "only the core contributors should have a say". If social order in communities is considered important, there should be well-defined processes on how to gain access to those status groups with in the community that carry weight in important decisions.

FOSS communities sometimes discount the importance of decision making, or claim that decisions are made or conflicts resolved by "the wider community", and that therefore organisational theory does not apply. This argument however does not hold, since it cannot be reasonably disputed that communities delimit members from outsiders, have status groups, make decisions and resolve conflicts (even if those elements are not all made explicit).[44] By analysing organisational structure, decision making processes and the community social order as key elements of governance, it is possible to compare communities even if they create unrelated products.

2.2.1 Growth stages of communities

The differentiation between the inside and the outside view of the communities social process puts the emphasis on the demarkation of the social group, in as much as it defines who is a member of the group and who is an outsider. Being an insider means accepting the group's rules, providing influence and in turn expecting to participate in the group's governance. Being an outsider leaves a choice of interacting with the community and accepting its norms, or to abstain from interacting with it. The community is afforded the same choice not to engage with an outsider based on how compatible their actions are with the group's norms.

Since about 2010, participation in FOSS activities as a phenomenon has changed from an exotic movement to a common mode of operation in the ICT industry.[87] This suggests that communities also have matured into established organisations with solidified cultural norms and values. The communities studied in this report have all existed for longer than a decade. They will be viewed as mature and stable organisations where processes can be observed through the activities within their formal and informal structure. Their norms and values have developed over time as a result of the interaction between community participants who join the group voluntarily out of their own motivation, and the community as an organisation of its own, which creates structure and processes according to the goals of the group and the strategies chosen to reach them.

The communities will be investigated at three different growth stages: The time of foundation called the *initial stage*, the time when the group has reached a small to medium number of contributors (typically between 20 and 50 active contributors) called the *medium stage*, and a *late stage* with a large number of community members (often more than 100). The growth stages are defined by the coordination mechanisms applied to the social process, which show different characteristics in these different stages of development.

At the time a particular FOSS initiative is formed in the initial stage the goals and motivations of the group of founders and the initiative as a whole are identical. There is commonly great enthusiasm about the joint initiative. The original authors publish their work, and in some way communicate that contributions from others are welcome and appreciated.

More contributors join and will participate out of a motivation similar to the one of the original authors - contribute to the product, make it available to the public under a FOSS license, and rely on the community to keep the process going. As long as the group is small enough for *ad-hoc coordination*, the contributors joining subsequently will find themselves in a similar situation. It can be assumed that the participants in the initial stage will be homogenous in their motivations, cultural backgrounds and interests. Worries about governance usually do not exist.[120][80][154]

Interests and motivations will start to diverge as the community grows and matures. The group will reach the medium stage when the number of participants becomes too large for ad-hoc coordination and changes into a form of consensus focused auto-organisation. In this stage, deviations between individual expectations and community behaviour exist. Instead of relying on formal structure in the organisation, the communities rely on a consensus-driven, participative debate culture. Disagreements will be discussed at length until a resolution is achieved. The resolution does not necessarily require consensus or a formal decision. The KDE community, for example, applies a method called “lazy consensus”, in which contributors already work on their favoured solution while alternative courses of action are still being discussed. The direction the community later prefers can then be decided based on the results of the discussion and on the experience from the work already provided by the contributors. Other communities apply similar mechanisms that prefer product related contributions over “bureaucracy”. It is apparent that such mechanisms rely on close cohesion of the group’s participants, a low grade of specialisation amongst the contributors and a relatively small number of stakeholders in the decisions. Not only are the communities content with such informal self-coordination, they also develop a strong preference for the absence of formal structure. Since contributors participate voluntarily, they feel entitled to self-identification of tasks and to work free from direction given by others.[51] While it may cause friction, self-identification contributes to the allocation efficiency of peer-production processes.[11]

The transition into the late stage of community development is commonly marked by more formalisation. Communities may establish internal working groups to facilitate contributions to more specialised topics. To coordinate with external partners, they may nominate community members to represent the community in their committees. To account for these delegated responsibilities, the representatives may be required to report on their work at regular assemblies. In general, more functional differentiation occurs between the community participants. Delegation of power and responsibility becomes more pronounced, leading to a more prominent role of the community leaders. Now leadership positions that previously more or less fell to those who volunteered to speak to the press or be elected to the board become more prestigious. Appointments carry more weight and elections for them grow competitive. The differentiation of roles within the community enables jockeying for position and a sense of entitlement, especially as regards long-standing contributors. Once this formal organisational structure is established, the community shows behavioural patterns similar to other larger common good oriented community organisations like unions, sports clubs or cultural initiatives. Being a part of the community becomes a motivation in its own right, complementing the motivation to contribute to the community product directly. Matters of procedure and community management attract more attention. A share of the collective energy of the community is redirected inwards to discuss the community itself. At the same time, behavioural norms are still in place that developed during the early and medium stage. For example, communities have established a "break all the rules" rule that postulates that every participant is free to decide the best course of action, even if it means ignoring a norm or rule. Or there may be a "who does the work decides" rule which postulates that those who take part in the debate should not interfere with those working directly on the community product.

Based on these considerations, it can be expected that communities in the initial stage require almost no coordination, communities in the medium stage rely on organic self-coordination, and communities in the late stage act more in accordance with the logic of collective action in large groups.[93] The transition into the late stage should necessitate a change of the effective community governance norms away from informal mechanisms of the medium stage towards more explicit, formal mechanisms appropriate for larger-scale collective provision processes. The intense governance-related conflicts and debates that accompany the shift of the communities into the late stage indicate that this change did not fully happen in the cases studied in this report.

Intense inner social conflicts indicate a divergence between the individual expectations of contributors and the group norms developed by the community. These conflicts may be resolved positively, resulting in a re-alignment of individual and group motivation. However, if the conflict is too severe or for other reasons cannot be resolved satisfactorily, it may also lead to either individual contributors deciding not to participate in the group anymore, or the conflict may cause a fork, where the group splits into two that continue to develop towards the initial goal separately.[125] Forks are rare, as substantial effort must be invested to create a competing community organisation. More commonly, contributors defect if the perceived quality of the community diminishes. Since there is no centralised resource planning, defections may go unnoticed. It is difficult to assess the impact of individual decisions or the design of decision making processes on the contributor base. Sometimes communities prefer not to make any decisions to avoid losing contributors, which results in indecision manifested for example in bike-shedding debates.[51]

2.2.2 Community composition

Entities participating in FOSS initiatives can be either individual volunteers, organisations (participating directly or through contributions of their employees) or staff employed by the community. This mix is referred to as *community composition*. Most communities consist of individual volunteers and employees of businesses, with a very small share of employed staff.[123] This study focuses on communities that are made up predominantly by individual volunteer contributors. These communities, like KDE, distinguish between contributions of time and effort spent by an individual contributor and the contributions made by businesses. The individual contributors are able to become personal members in KDE e.V. even if they contribute during work time. The businesses employing them may only gain "supporting membership" through which they support funding the organisation by paying a membership fee, but do not attain a vote on a board or in the annual general assembly. FSFE similarly does not allow other organisations to take part in their activities directly. These rules underline the significance volunteer driven communities associate with individual contributions and the aversion against any form of institutional investment. The norm of exclusively valuing individual contributions builds upon the expectation that the community production process should be steered with regard to product quality alone, and not influenced by interests of external parties. In the case of KDE e.V., this norm is explicitly codified in the bylaws of the organisation, which only accepts individuals as members, not legal entities.

In communities with a majority of contributors employed by businesses, like the Linux kernel developer community, the reputation of companies is more closely related to the aggregated contributions of their employees.[36] Businesses and individuals participate in FOSS activities for different sets of reasons. Individual volunteers are mainly intrinsically motivated through a sense of achievement and personal enjoyment. Signalling of key skills to potential employers also plays a role.[80] Businesses are motivated by economic rewards. For example, participation gives them the opportunity to provision non-differentiating components to their products in collaboration with other parties with similar interests at drastically reduced research and development cost as well as participation transaction cost.[88] Businesses also benefit from their FOSS activities being a source of quality staff and promoting a healthy innovation ecosystem.

Depending on community composition, the communities develop norms and principles that reflect the specific mix of motivations of their constituency. This opens up a continuum with purely volunteer driven communities on one end, purely business driven communities on the other end, and mixed or hybrid communities in between. The majority of FOSS communities are hybrids, resulting in a set of norms and practises within the community that reflects the motivation of both organisational and individual contributors.[131] We expect that the norms and principles adopted by the communities can be clustered based on the contributor composition, and that communities with relatively similar contributor structure develop relatively similar norms and practises. To facilitate separate analysis of these sets of motivations, this paper focuses on studying communities that are (almost) exclusively made up of individual volunteer contributors. These communities should then have developed comparable governance norms.

2.2.3 Separation of open source products and community processes

FOSS communities create freely available *products* in a social *process* of peer production.[11] While it is a common expectation that producing a product under a free software license goes hand-in-hand with applying a transparent, open process based on voluntary participation, this is not always the case.[84] There are FOSS products that are produced by a single vendor in a closed process and without relevant participation of other parties.[122] Some products are developed by a single, dominant commercial vendor where outside participants are required to grant rights to relicense the product proprietarily to the commercial vendor through some form of contribution agreement. These agreements do not reduce the freedoms provided by the product license, but they change the community process from decentralised to centralised.[51] Other products, like the Linux kernel, are built by a decentralised community and do not require any attribution of rights. The licensing of FOSS products on the one hand and the community processes applied to produce them on the other need to be considered separately. While the choice of license defines whether or not a product is free software or, synonymously, open source, the governance norms applied by a community determine openness.[79]

We assume that the preference in a community for a more or less open governance model correlates closely with community composition (2.2.2), and that volunteer driven communities have a strong preference towards openness and transparency in their governance.

The two main schools of thought about the essence of FOSS represent these two aspects separately as well:

Some proponents of the term "open source" put more significance on the fact whether or not a product is distributed under a FOSS license approved by the Open Source Initiative. They see software released under a free license as a means to an end. Others who put more emphasis on software freedom consider the work of communities to be part of a political movement representing a cultural shift that works towards a world without proprietary software, with an ethical underpinning. The FSF for example argues that "software should not have owners".[140] The separate product and process aspects of FOSS however are present and relevant in both schools of thought.

2.2.4 Voluntary participation and meritocracy

Both camps agree that contributors form the community by taking part in the production process voluntarily and without direct compensation for their efforts. Communities with a small number of contributors are typically organised in an informal way and rather coherent. As the number of contributors grow, the difficulties of informal organisation grow until they reach a level that requires more formal structure. There is however no authority that is in a position to impose such structure. The raison d'être of enterprises and institutions is commonly defined ex-ante by for example investors or the government, who act as a form of higher power that imposes purpose on the entity. Similar to sovereign states, the question for the purpose of FOSS community is self-referential. A community exists to serve the interests of the participants, who also are the community. Where states resort to postulating a constitution which then anchors acts of government, communities develop governance mechanisms based on voluntary participation and meritocracy (unlike in the original satirical understanding of meritocracy, the wider open source community uses the term with an overall positive connotation).[161]

Voluntary participation in the group means that acceptance and implementation of group decisions needs to be achieved based on a common mutual understanding. Since they are contributing voluntarily, participants expect to have peer status in the group and influence according to the value of their contributions. This is what FOSS communities refer to when they call themselves meritocratic. When formal structure emerges, the principles and norms applied typically reflect voluntary participation and meritocratic peer status as well. From this derive two important collective action issues that communities struggle with, that of decision making, and that of enforcing conformity to social norms.

Decision making is difficult as the winning majority has no instrument to force those who disagree to implement the decision. There is no individual cost involved in simply ignoring a community decision. Communities therefore prefer to reach consensus in an elaborate discussion process over a decision even with a very qualified majority. They purposefully refrain from concluding a debate after a decision was made by allowing the same question to be re-raised without much effort. Some communities explicitly acknowledge the difficulty of making formal decisions and relegate them to the status of opinion polls (Wikimedia) or restricting the use of votes to the acceptance of new members (KDE). The sensitivity regarding making decisions that are not based on consensus reflects the importance of attracting and retaining contributors and underlines the social process aspect of community work. This sometimes results in a separation of administrative leadership and product related decision making. For example, KDE e.V. manages KDE's assets and funds, but by way of a community principle may not interfere with product related technical decisions.

Mechanisms that aim at enforcing conformity to social norms are mostly absent in communities. Initially most behavioural norms develop informally. In the medium and late stages, community manifests and a code of conduct may be put in place. Already then, the necessity of a formal rule that restricts how community members may behave was questioned, for example in KDE and FSFE. The studied communities did not build effective means of actively influencing behaviour towards the expected. While in early stages this is mitigated by the strong cohesion of the group, in later stages the lack of it is often seen as an obstacle to developing more diversity.[2] Based on anecdotal evidence from the interviews, the necessity of explicit behavioural rules is typically questioned by longstanding community members that are part of the dominant social group within the community. The aversion to enforcement of rules is related to the self-referential nature of communities. Critics of explicit rules often question where the authority would come from to enforce them.

Governance based on voluntary participation and meritocracy is an essential attribute of volunteer-driven communities that stems from the self-referential nature of the community's purpose. While communities exist that are led by a "benevolent dictator", the cohesion of these communities depends on this individual maintaining a strong merito-

[2] http://rachelnabors.com/2015/09/01/code-of-conduct/ (accessed 03/02/2024), but also https://modelviewculture.com/pieces/a-code-of-conduct-is-not-enough (accessed 03/02/2024)

cratic status.[120] Separating product aspects from community process can also provide a guideline regarding which FOSS producing organisations should be considered communities: a community distributes products that through applying a FOSS license are public goods *and* the organisation needs to create this product in a process that is based on voluntary participation. When applying these criteria to classifying organisations into FOSS communities and others, the cases studied in this report match, while a single entity producing a free software product without voluntary participation of others, like Google producing Android, would not.

2.2.5 Study design and method

The aim of this study is to describe and understand in detail how and based on what influences governance norms in volunteer-driven FOSS communities emerge, and what effect these norms have on the community as it grows from a new initiative to a large size organisation. A qualitative embedded multiple-case study of the inside view on social norms in three communities was performed. It may be considered a mixed method study design that combines the multiple-case study with theoretical modelling based on personal observation and experience, however that personal experience is also embedded within the same cases.[160]

The cases analysed in this study are large, mature, successful volunteer driven FOSS communities (see page 42). There is only a small number of communities that achieved such success over an extended period of time, probably about a dozen. Another key criterion for selecting the case studies was access to individual key community actors and the organisations' important decision making bodies. The author has access to internal information of some communities because of his own history as a long-term contributor. Since there is only a small number of communities that reach the late growth stage, and these communities develop a strong cohesion and a distinct insider culture, the expectation stands that an analysis from outside the organisation will not lead to a significant improvement of understanding how the communities function. The qualitative study design accounts for the interpretive, experience based, situational character of the cases, and facilitates analysis of organisational development as a long-running, episodic, evolving phenomenon. A small number of community cases was chosen also to avoid stereotypical generalisation caused by an unwarranted higher level of aggregation.[139] The decision in favour of a qualitative research approach was supported by the assumption that quantitative methods do not promise reliable insights given such small constituencies. Experiments also were not considered feasible.

The study was conducted by performing 16 interviews with longstanding contributors to the communities analysed in the case study that where either founders or later rose to community leadership positions. Some of them are still active today. Some resigned from their functions. Overall, the interviewees that contributed to this study represent more than 200 person-years of FOSS community leadership experience. The interviews gathered information about the personal ethics and convictions of the contributors, and about their interpretation of how the community governance norms and organisational design have developed.

The interview concept was developed against the theoretical framework (see section 2.2) which was built upon the individual experience of the author and the current state of FOSS community governance research. The interviews provided evidence that puts the experiential expectations into perspective.

The interviews helped obtain unique interpretations held by the interviewees, to aggregate information from many interviews, and to access the personal experience of the interviewees that is otherwise unobservable. The interviewees provided data about their own individual positions, and about their interpretation of questions about the community as a whole. By connecting observational insights with understanding of the individual ethics and convictions and observations on community governance by the interviewees, expectations in the theoretical framework about community governance norms and contributors expectations can be tested.[139] The final report is based on personal observation, the interview results, and information available in artefacts like minutes from general assemblies, statutes, manifests or codes of conduct that the communities published.

The qualitative method chosen leads to limitations to the applicability of the study's results. The emphasis on the inside view regarding governance norms does not take external factors like the explosive growth of

FOSS into account that may also have contributed to community growth and other development trends. Demographical changes affect the communities - one interviewee mentioned a perceived decline in the proclivity of individuals to volunteer for social causes. Market trends that affect the position of the community's products also probably play a role. More importantly, the subjective, personalistic, constructivist point of view applied in the study means that observations only represent the experience or interpretation of the participants and the author, not necessarily a true meaning. The findings in this report can therefore not be generalised. They should however provide a valuable deep understanding of the inside view the communities and contributors have on themselves.

2.3 The mindset behind community governance

The interviews for this study consisted of three separate parts. The first one focused on the individual contributors, what their expectations and convictions where and why they joined their communities, how these expectations developed or changed over time, and what principles or ethics of individual conduct are important to them. The second part of the interview focused on the community as a whole, and the third part discussed inner-community conflicts as focal points for governance debates. This section is based on the first part of the interviews. Some of the key governance documents like organisational statutes, the codes of conduct or community manifests have been authored by the interviewees. It is assumed that since the interviewees are founders or long-time participants in the communities and through their leadership roles actively influenced the community constitutions, their expectations and convictions strongly influenced the emerging governance norms. Even if these may have changed at a later point in time, their influence should still be noticeable.

2.3.1 Engage in a community of makers

It is commonly assumed that participation in the development of FOSS products is primarily need-driven.[120] However, the need for a solution to a particular problem does not explain sustained investment of effort into being a community member in good standing. To justify this behaviour, being part of a community must generate additional rewards like a sense of belonging. The most limiting factor to contributors is the time available for such intrinsically motivated work. The different projects compete for this time.[80] Being an integral member of a community means to share time available between creating contributions to the product, which is perceived as a fun, productive and creative activity, progressing through a social hierarchy and engaging in community processes, which may be considered a necessary but time-consuming overhead. The initial question to answer is why contributors that end up being involved over an extended period of time and investing significant amounts of their personal time into FOSS contributions make the decision to join the community in the first place.

All participants in the study stated that the impetus to engage with the community and become a part of it was to contribute to the community's main product. They also stated that to create that product needs to be a positive challenge to be motivative, not a routine task. Only a third initially decided for a specific community because of its social norms.
A strong majority however said that over time, being a part of the social group became more and more important to them. A common phrase used to describe this phenomenon is "come for the technology, stay for the people." The freedom to choose what task to work on in a group of like-minded people and the creativity in the search for a solution that this affords was mentioned by almost all interviewees as a motivator for becoming part of the community. This indicates that contributors are attracted to a FOSS community because of the challenging products it sets out to create, and only then learn about the social norms within the community and begin feeling attached to them. Those interviewees that are not active in their communities anymore usually exited gradually, reducing the amount of their contributions over time until they stopped. The expectation of individual productivity is in line with existing research which identified the felt personal sense of creativity as the biggest impact on contributed hours.[80]

Multiple interviewees mentioned that they felt the community mission was "worth fighting for" in that it combines a productive, creative activity with a sense to contributing to a greater good, like fostering technical innovation, building up competition to dominant proprietary products or advocating for the societal benefits of software freedom. The communities provide a virtual place where individuals who share this combination of rather specialised concrete need and ethical conviction congregate. In cases where this overlap of interests is rare, a locality may not reach critical mass to become a gathering place for like-minded people.

Communities are meritocratic in the sense that individuals gain influence solely based on their contributions to the combination of community product and social process. This environment attracts highly skilled individuals that interviewees felt they could look up to and learn from, but at the same time accepted them as equals. Such learning is a rare opportunity not commonly available to highly skilled individuals in physical environments. Meritocratic peer status based on concrete contributions also leads to a notable absence of other forms of discrimination by for example nationality, race, gender, age or other factors, at least initially. Individuals with non-binary sexual orientation are a common sight at community events, and do not usually attract much attention. One interviewee assumed a higher-than-normal share of individuals with symptoms of autism or Asperger syndrome amongst the contributors.

2.3.2 Equality of opportunity among peers

The interviewees joined their communities when they where still in the initial or medium stage. Some explained the perceived group size as “tiny”, or mentioned that there was a positively motivating David-vs-Goliath feeling to working towards the group’s goals. Most explicit and implicit governance norms where established in these phases. If communities need to provide a combination of productive contribution opportunities and matching ethical convictions, what would be the expectations the contributors had when they joined towards how the communities should operate?

Only a minority of the interviewees joined their communities with expectations towards their governance norms. Some of the community processes in fact came as a surprise to newcomers, for example the extend to which new participants where immediately accepted into the group and even encouraged to represent the project at community and other events.

Some intentionally joined the community to learn about how it works, and stayed in an observer role for a period of time. Most participants developed their preferences towards governance norms while being a contributor.

Most of the interviewees emphasised a priority of "doing" over "talking". Contributors to the KDE community are very conscious of the "who does the work decides" rule. While an absence of discrimination is generally expected, the meritocratic rule within the communities does not translate to egalitarianism. Participants earn their prestige or even the right to participate in debates within the community through the contributions they make. This translates to an expectation of *equality of opportunity*, but not of an equality of rights. Ideally, the status attainable by any contributor only depends on how much effort she or he invests into contributing to the community's causes. Community members who "only talk" found little acceptance and where sometimes explicitly denied a voice in debates. Some stated that initially they felt like the community needed no other structure than grass-roots meritocracy, but that in later stages they changed their mind about that.

Almost all interviewees mentioned an inherent tendency within the communities to form sub-groups that specialised on particular functions or product aspects. These sub-groups remained at a size that continued to allow for ad-hoc coordination, even as the overall community grew beyond a size where this would be effective. The governance within these sub-groups was less standardised, one interviewee described them as "little villages with chieftains". Sub-groups also helped to retain regional or cultural cohesion and the sense of productivity by isolating their members from what some described as excessive debate. Because they initially associated themselves with one of the sub-groups, the communities felt smaller to the interviewees at the time they joined than they really were in numbers of overall participants.

Surprisingly, the fact that the community product is distributed under a free software license or generally is a common good was not mentioned as an expectation by the interviewees, but as a precondition. Similarly, the absence of discrimination is expected as a given. Some said they would simply not consider participating in any initiative unless the outcome is freely available to all.

2.3.3 Balance of makers and community builders

The communities in this study all succeeded in establishing themselves as important in their respective fields and grew from the initial into the medium stage within two to four years. Almost all interviewees mentioned that being a member of the community became a goal in itself, where previously community membership was a means to facilitate contributions to a product. The contributors build personal attachment to the community as a sort of virtual home, where they maintain friendships and develop loyalty to the group. Some of the early contributors quickly rose to community leadership positions that became an important part of their self-perceived identity. They reallocated a share or all of their available time to community management tasks, reducing their product contributions in the process. Differentiation emerges between the product developers as the makers and the community builders as the maintainers.

Most participants could not rely on previous experience in managing larger communities and partly were surprised by their own success. One interviewee involved in Wikimedia explained how the explosive growth of the community in 2004 caused the group to consider how to coordinate "once we were more than three". The communities struggled with the transition out of the initial stage. For multiple interviewees this transition happened when they realised that they did not know all contributors personally anymore, which also indicates a breakdown of ad-hoc coordination. The KDE core team retreated into private invite-only mailing lists where "those who do the work" could coordinate. The need for organisation and administration manifested itself when the community started to organise the first all-hands conferences. To manage funds and donations the legal entity of KDE e.V. was created. The reliance of private communication channels was felt to contradict the expectation of transparency and open process by one interviewee. However, it was considered necessary at the time, and is still in place today.

Contributors that mainly work on community building face a dilemma that only becomes apparent over time: while they are contributing to core functions of community management by being board members or project representatives, they are not taking part in product development anymore in a community that primarily exists for that purpose. Consequently, their merit eventually declines. Some cling to their influential positions, possibly realising that they will not be able to maintain their community status once they hand over to a successor. Progressively they disconnect from the product-focused elements of the community, and begin to value procedural questions about community management higher than facilitating the productive processes. Instead of being supportive, the administrative entities created by the communities exhibit a tendency to grow to be *antagonists* to the community of makers, "with members who contribute little and a board that contributes even less", as one interviewee described it.

2.3.4 An ambitious, productive meritocracy of equals

Over the years that the participants in the study contributed to FOSS, it can be assumed that it is common for them, once in a while, consciously or sub-consciously, to take a step back and reconsider whether their time and money spent towards the community purposes is still a worthwhile investment. We asked them what criteria they apply when evaluating the perceived quality of their community. The answers were surprisingly uniform across the participants from all three communities. All or almost all interviewees agreed to the following criteria:

The communities need to provide a *welcoming, inviting culture*. It forms the basis for the close social connection that develops between contributors. The communities should also extend trust to newcomers, affording them to learn the community norms even if it involves making mistakes. This includes flat hierarchies for contributing to the community products. The communities implement an "open doors policy", as KDE puts it, where newcomers, once they have an account, have access to almost all of the project's infrastructure. Common well-accepted exceptions are system administration, legal and financial functions.

Participants expect their communities to implement *meritocracy*. While the understanding of meritocracy is not completely uniform, mainly regarding what constitutes a contribution and how it should be valued, the prestige and influence of contributors within the community should be measured by the aggregated value of their contributions, and nothing else. Two aspects of meritocracy are less defined in this expectation, how merit decays over time (forcing old-timers to make way for new contributors), and which other soft factors like socialisation, being in the right place at the right time, gender or age influence merit. More recently, *liberal contribution policies* as applied by the Node.js project address these issues.[126] Experienced contributors consider meritocracy in FOSS communities very important and successful, but question the naïve understanding of it that is commonly applied.

Related to meritocracy, there is an expectation of *equality of opportunity*. All communities, even if they down-play it, form status groups like administrators, formal members of the organisations or elected positions, with noticeable barriers to entry. A common expectation is that all these positions should be open to anybody willing to contribute enough to the community cause subject to a common set of criteria. Equality of opportunity is different from meritocracy in that status and merit may diverge. Individuals that attained status based on past merit that has now decayed may still be influential in the organisation. Similarly, a valuable contributor may have merit but not advance in status because, for example, no elected positions are available or old-timers get elected to them. This may cause personal disappointment, possibly disgruntle valuable contributors and eventually cause them to spread negativity or leave the community.

Contributors are looking to contribute to *useful, productive communities*. They want their contributions to help the community to get closer to achieving its goals. It is often not enough to contribute to the product, contributors also expect the product overall to be useful, and to receive feedback or even to get more contributions from users outside the community. One of the main values that the community adds to the peer production process is to add distribution and communication channels to attract users to the community products and create a feedback cycle back to product development.[11] An increase in the required share of time available spent on community-internal debate detracts from the sense of productivity.

On top of the community helping them to be useful and productive, contributors expect the community's mission and vision to be *ambitious*. It is not enough in the long term to "build a better mouse trap", as the intention to build a FOSS replacement for a proprietary product has been described. Achieving societal change towards software freedom by lobbying for it is considered an ambitious goal, as is freeing a large user base from lock-in to proprietary products, or creating "a world in which every single human being can freely share in the sum of all knowledge", as in the Wikimedia vision. In an abstract sense, contributors want the community to aim at making the world a better place, and their contributions to help with that.

Next to these four quality criteria, the interviewees mentioned aspects that pose preconditions for engaging with a community. These preconditions may be considered *hygiene factors*, criteria that do not positively motivate contributors, but whose absence would be considered a reason not to engage with the community at all.[64] Such factors include that community products are public goods, an absence of discrimination, a positive communication culture or code of conduct, respect for minorities, reasonable escalation mechanisms, supportive technical infrastructure, and opportunities for learning and personal improvement. These factors are "basics that need to be there".

We asked the participants if they felt a sense of responsibility for or *loyalty* to the community as they progressed, which they unanimously agreed to. Some felt that the team they worked with started depending on them, and even tried to empower their colleagues to reduce that dependency. It would have felt bad for them to leave the community while this dependency existed. The merit they attained and the personal relationships built with other community members gives them a sense of responsibility for the community as a whole. They also understood that it would be hard for them to replicate the time and effort invested, which imposes a cost on exit that makes it difficult for long-term contributors to leave the community.[65]

The communities struggle with staying inviting to newcomers. Both product- and process-related barriers emerge and grow over time. In the initial phase, all contributors are beginners, and mistakes are commonly seen as part of the process. Later, longer-term contributors have accumulated experience, and the quality gap between contributions by them and by newcomers widens. The German language Wikipedia for example introduced a pre-publication review ("Sichtungsprinzip"), which reduces the trust extended to new contributors. This reduces the feeling of appreciation and acceptance that the participants reported they themselves felt when they started contributing. Interviewees said that the trust that contributions in average are of positive value has been lost to some older community members.

2.3.5 Ethical principles applicable to community governance

Contributors will only be intrinsically motivated to voluntarily spend significant efforts in a social group that conducts its activities in a way that agrees with the ethical convictions and principles of the individual. Such convictions are formed through life and rarely change. They can be considered an external variable that the governance norms of the community need to reflect. We asked the interviewees which principles that are considered "just" in other social groups, they think, should also be applied in their communities.

The interviewees strongly agreed that working code, meritocracy, solidarity and transparency are key principles that they look for in the governance of their community. *Working code* refers to the expectation that "code should speak louder than words", meaning that concrete code or other contributions to the community product should be valued higher than "politics". This argument is related to a general paradigm that postulates that FOSS development should focus on delivering a working implementation over, for example, writing a detailed specification.[154] They feel that the work on the product should be the benchmark by which the community is judged. This principle is important to contributors because it describes very directly how the communities should operate. *Meritocracy* is mentioned again as an individual expectation, indicating that the term is not only used to describe a mechanism of community management, for example in codes of conduct, but also as an expectation for a norm that directly influences the motivation of individuals to contribute. Communities implement meritocracy because their contributors expect them to or would otherwise not participate. *Solidarity* is a principle that shows itself in an extend of trust to newcomers and more experienced contributors, a believe in the generally good intentions of them, and a habit of mutual support. It is the part of the fabric of the social cohesion that the communities form and enables them to overcome otherwise separating attributes like race, gender, nationality or age. Tensions in debates have often been resolved in good humour by invoking Hanlon's razor, reminding everybody involved not to attribute to malice what can be adequately explained by (collective) stupidity. *Transparency* is a common expectation that should result in processes and debates that are accessible equally to and documented for all contributors. This is understood as an invitation to participation, not a duty. The transparency principle is to a large extent engrained in the technical infrastructure of projects. Discussions take place on mailing lists, wikis or online chats, and are commonly logged or otherwise preserved. Activities are coordinated in project management tools or task trackers, often in ways similar to how a software development project would be organised. This habit may be encouraged by familiarity with software engineering tools. Interviewees from all three communities mentioned that they feel like their organisation is not as transparent as it should be with regard to governance processes, as opposed to product contributions.

There is no agreement on whether or not communities eventually need to fall back to *majority decisions*. Some interviewees believe that if the community cannot reach consensus on a subject, it is better if no decision is made at all. Others accept that situations exist where making a decision is inevitable. When asked directly, most but not all would prefer a decision over non-decision. All understand that with voluntary participation decisions cannot force contributors to act in a certain way. However, there was also disagreement over whether or not consensus should be sought as the decision making principle. There is little awareness that for many issues, staying with the status quo is one of the alternatives to choose from, and that *not* making a decision is equivalent to deciding to stick with the status quo. The consequences of decisions are being evaluated, those of indecision commonly are not.

The expectations regarding decision making processes appear to be changing over time, with multiple interviewees reporting that during their early involvement their preferences have been leaning much more towards unstructured ad-hoc coordination, while after being involved for a number of years they feel that better defined and documented decision making processes and especially escalation and conflict resolution mechanisms would be necessary. It is not clear if this change is caused by gaining more personal experience, or by the communities outgrowing the initial and medium stage and now operating as larger organisations.

The concept of transparency is connected to a number of common expectations about what constitutes a FOSS development process. Habits like open technical discussions, online collaboration and releasing working code early and regularly are considered strengths of the wider open source community. To enable that, a contributor or newcomer should be able to understand what the community is working on and how to take part in it based on information available online. Contributors also need to ensure that they possess all rights to use, study, modify and distribute the community product without a need for later negotiation. The emphasis on transparency is born out of the necessity to facilitate distributed collaboration in a diverse team.

There is an understanding that the communities implement these principles well with regard to the production processes, but not regarding community governance. Especially a lack of transparency and meritocracy is noticeable regarding the decision making of the community leadership and staffing high ranking community functions. In terms of documented structures and processes, the communities do not differentiate between product and governance related decisions, even though many of the norms applied rely on the fact that technical changes can easily be reverted.

The interviewees mentioned that there is a close match between their personal ethics and convictions and the social norms they expect the communities to develop. The fact that the governance of the communities is modelled so closely after their ideal of how an organisation that benefits the common good should operate is a strong motivator for them to continue contributing.

2.4 Case studies

The communities studied for this report are all primarily volunteer driven (their contributors are *amateurs* in that their community engagement does not constitute a significant direct source of income, as opposed to professionals), mature (they have been working towards their purpose for multiple years), comparatively large (they have attracted between dozens and hundreds of contributors over time) and successful (each of them is recognised as as an influential organisation in their respective field). Even though all of them produce freely-licensed public goods, the communities differ in the nature of their main product: the KDE community primarily produces software with a focus on end-user needs, FSFE is a free software pressure group that advocates the benefits of software freedom and Wikipedia produces an online encyclopaedia. The community product is the key element that provides participants with the opportunity to contribute. However the communities have been selected using the hypothesis that community composition has a more dominant impact on governance norms as they emerge than the nature of the main community product.

Governance norms are expected to primarily develop according to the expectations and convictions of the contributors regarding the characteristics of the collaborative peer production process. The following chapter analyses the vision and mission, the formal and informal organisational structure, the decision making and conflict resolution mechanisms, and the rules for group membership of each of the studied communities.

2.4.1 FSFE

Mission, foundation and history

FSFE was founded in 2001 with the mission to bring about sustainable change towards societal freedoms in the use of digital technologies. The ambitious "life-time scale"[3] of this mission was understood as comparable to initiating "a second enlightenment" with regard to software freedom in Europe. At the time of foundation, FSFE was considered the European sister organisation to the FSF. The organisation gained recognition from representing the wider open source community in the anti-trust case of the European Commission against Microsoft's dominant position as a supplier of operating systems for personal computers. FSFE also represented the community at the World Intellectual Property Organisation and the Internet Governance Forum.

FSFE introduced a fiduciary licensing program in 2003 that allows

[3]Quotes in this section are taken from the interviews, unless otherwise noted.

FOSS contributors to have their copyright interests represented by the organisation. The fiduciary licence agreement (FLA) program strengthened the role of FSFE as a representative organisation of the European free software community.

In 2005, FSFE launched its fellowship program, widening the base of supporters to those who wished to contribute to the organisation's purpose financially, instead of by investing personal time. The fellowship had a limited representation in the general assembly through two seats for fellowship representatives until 2018, when the fellowship program was terminated.

With the increasing adoption of FOSS in commercial products, the complexity of compliance with the free software copyright licenses and the danger of free-riding behaviour of some manufacturers became apparent. With support from external parties and in cooperation with the gpl-violations.org project[4], FSFE launched the Freedom Task Force, an initiative intended to help contributors and businesses to create and use software under free software licenses correctly. Related to the Freedom Task Force, the Legal Network was founded as a venue for legal and technical experts to collaborate on legal and licensing issues related to free software. The Legal Network is the single largest network of free software legal experts world-wide.

FSFE continues to grow in influence and size. Today, it wields relevant political influence at the European and EU member state level, has a strong backing by the FOSS community, and provides the most influential venue for legal and licensing discussions globally. It employs a president as well as a small group of policy analysts, campaigners and administrative staff.

FSFE offers opportunities for FOSS activists to participate in a small set of well-defined key products - political influence on the regulatory framework relevant for free software, coordination of various regional free software related activities, and facilitating the resolution of free software legal and licensing issues.

[4] The gpl-violations.org project (`https://gpl-violations.org` (accessed 03/02/2024)) tries to raise public awareness about past and present infringing use(r)s of GPL licensed software.

Formal and informal organisational structure and conventions

The original intention for FSFE's organisational structure was a federated system of regional chapters with a central coordinating office that represented the organisation at the European and global level. Based on subsidiarity, the local chapters would be autonomous except when central coordination is needed. The concept of local chapters did however not materialise, with only one ever being active. The idea of local chapters was eventually dropped in 2016, leaving the organisation with a headquarters in Berlin that represents FSFE across Europe.

The legal entity of FSFE is an "eingetragener Verein", a charitable association registered in Germany. The statutes are the most explicit documentation of FSFE's structure and processes. The formal members of the organisation are somewhat misleadingly called the *general assembly*, even though it is a permanent decision making organ. The general assembly elects the president and vice president. Historically, the president has been the most visible and influential role in the organisation. The president, vice president and treasurer together form the executive council, the de-facto day-to-day decision making body of the organisation. Activities are coordinated between the FSFE *team*, the group that comprises all active contributors with separate communication channels and a formalised, email-based decision making process. The general assembly and the staff are subsets of the team. Occasionally, task groups are set up to handle specific topics.

Informally, especially in the initial stage of FSFE, some individuals exercised significant influence without a formal mandate, called "luminaries" by some interviewees. Since FSFE was founded as a sister organisation with the "blessing" of FSF, early activities where coordinated with FSF, and approval was sought for key political positions and messaging. The influence of FSF waned over time, also because FSFE applied a more collaborative style of governance than FSF. Approval towards general assembly membership is handled very selectively. There are no term limits or requirements for re-election for general assembly members. Early-stage participants still wield significant influence in the general assembly, even though some would not be considered part of the *team* today since they are not actively contributing. The approval for full membership in the organisation is selective and depends on a combination of individual initiative and pull from existing members. As of May 2017, there have been 27 full members[5], with about two thirds actively contributing in the past 6 months.

To some interviewees, the formal organisational structure does not reflect reality anymore. They consider the loosely-defined *team* as the core of the organisation, since most of the day-to-day work is coordinated there today. The team however does not have authority over budgetary or executive decisions that are a prerogative of the general assembly, marking a significant deviation between power and responsibility. Many conventions are implicitly defined and passed on by word of mouth. Longstanding rules may still be in effect, but are not very well known or followed. Some norms and processes are clearly under-documented, which one of the FSFE founders during the interviews classified as a "rookie mistake".

Due to its history and initial community composition, FSFE has a regional concentration in Switzerland and Germany, with the head office being located in Berlin. Since almost all activities are conducted online, the impact of the community is spread relatively equally across Europe. Local (country) teams exist in 8 European countries as of January 2019.[6]

Decision making and conflict resolution

Most decisions are made at the FSFE team level in a consensus-driven, mainly email-based process. An issue is raised on the team mailing list, and after deliberation a proposal is submitted to a specific decision mailing list. The proposal is accepted if no rejections are raised. In the case of objections, the proposal is returned to discussion, refined and submitted again in an updated form. While this process could theoretically repeat multiple times, the consensus-driven culture within the group limits iterations so that almost no proposal reaches a third round of debate. If the team realises that consensus cannot be achieved, the proposal falls back to the president who then may abandon the proposal or, if considered necessary, force a decision. There is awareness of the need to find resolutions that are palatable to those who raised objections. The value of

[5] https://fsfe.org/about/team (accessed 03/02/2024)
[6] https://fsfe.org/about/groups.en.html (accessed 03/02/2024)

the decision is weighted against the cost of demotivating members of the team.

The general assembly decides based on a simple majority after detailed deliberation of an issue. Only a few more strategic decisions are made at the general assembly level. The local groups develop their own processes that are not prescribed by the central office. This approach works well since those groups are small in size.

In general, there is no defined way to appeal against a decision. Staff are supposed to direct complaints against decisions towards to executive council, in which the president is one of three members. There is no process to appeal against general assembly decisions. Overall, these convoluted and circular rules of appeal potentially result in an absence of accountability across the organisation. This is balanced by the dominant motivation to work towards a common goal, however there are no provisions against abuse. Compliance with norms and processes is effectively left to chance.

The decision making process within the team and guidelines as to how the general assembly and the organisation as a whole should work have been documented early on in FSFE. One of the interviewees assumed however that only a small fraction of those currently active in the organisation are fully aware of them. While staff and general assembly members may assume they are commonly understood, these documented norms and processes are not transparent to outsiders, and create a barrier for newcomers to participate effectively. As a result, reforming the formal structure has proven to be very difficult.

Community membership, roles and privileges

Throughout the time that FSFE represented the free software community in anti-trust cases in its initial stage, a significant risk of entryism was felt by participants. Formal membership in the organisation was made to depend on approval by the existing members, and applied selectively. This risk continues to be perceived as relevant by some in discussions about structural reform today. There are significant barriers to entry and a selective approval process to formal membership in FSFE. The vast majority of FSFE contributors are not formal members of the organisation. Governance is affected by this indirectly as the strong influence of long-term contributors or staff as members can be used to influence which issues are put up for a community decision. Multiple attempts at organisational reform in recent years ended in indecision.

There is an ongoing argument as to to what extent the formal structure influences the work of the community. While contributions to FSFE's mission do not require formal status in the organisation, the lack of clarity regarding ways to participate and access to key roles may have a detrimental effect on contributor engagement. Interviewees pointed out that contributors more commonly slowly fade away instead of leaving with a clear end to their engagement. This makes it difficult to measure levels of engagement, like the number of active contributors or contributions made in a time frame. The effect of the social structure on the success of the communities is not explicit.

Structural reforms and outlook

Similarly to the other case studies, the formal organisation of FSFE has only changed rarely. The fellowship program was introduced in 2005. The fellowship seats in the general assembly combined with the fellowship representative elections existed from 2009 to 2018, as did the role of executive director. There have been no other major governance changes to the organisation since 2009.

A *strategy process* was started in 2013 and is still ongoing. It was designed as a top-down process and mainly involves the inner circle of general assembly members and staff. The wider open source community especially outside of the regional focus area of German-speaking countries is not involved. While the process produced statements of intent, it did not influence the day-to-day work of the community much, or trigger an alignment of activities across the subgroups. This situation indicates a lack of forum for strategic discussions where all stake-holders engage, like an actual annual general assembly across a wider membership. The issue is exacerbated by the position of some members that FSFE is not accountable to the wider open source community, and only speaks for its members. An interviewee summarised the strategy process as "a lot of discussion and very little results".

According to the interviewees, there is no systematic process to maintain and document the formal structure and align it to the development of the informal one. Some rules are being ignored since the problem they anticipated, like a hostile takeover, did not occur. To newcomers the governance of FSFE is hard to understand and non-transparent. Contributors that are not part of the staff or the general assembly have practically no chance to influence the organisation.

An interviewee mentioned a perceived lack of impetus for change since about 2011, with FSFE's leadership mainly taking on a maintainer role (see section 2.3.3). This is exacerbated by a difficulty in creating effective collaboration between staff and volunteers. There is a chance that the contributions of hired staff displace volunteer work by reducing intrinsic motivation, creating a potential zero-sum scenario where spending changes whether FSFE receives contributions from staff versus volunteers, but less the overall level of contributions.

In summary, the formal and informal organisation of FSFE as well as its decision making processes appear well thought out originally, but have not been updated in pace with the growth of the community and outside changes. The well-thought out organisation design has aged and is now in need of reform. It appears that the main problem is not the perfection achieved with the initial setup, but with the absence of constant, gradual improvements to it over time. The KDE community took a different approach, but due to a comparable lack of a systematic improvement process, ended up in a similar situation.

2.4.2 The KDE community

Mission, foundation and history

The KDE project was founded in 1996 by Matthias Ettrich when he wrote to the `de.comp.os.linux.misc` Usenet group looking for contributors to a new, visually pleasing, easy to use graphical user environment for the more and more popular Unix operating systems, named KDE. Unix was regarded as the superior operating system to Windows, but the usability of modern graphical desktop environments on Unix systems was a real, widely-felt limitation at the time. The call for contributors fell on open ears in software development circles. There was a concrete need by early contributors to build the kind of desktop they wanted to use for themselves. The motivation of the founders and that of the initial stage community was identical. The possible achievement of competing with the dominant proprietary desktops produced by large enterprises was an important motivator. Within less than a year, a small group of early contributors produced a first version of the new desktop that showcased the enormous potential for innovation in this niche.

KDE released multiple successful versions of its main product, the "K Desktop Environment" until today - version 1.0 already in 1998, 2.0 in 2000, 3.0 in 2003 and 4.0 2008.[7] By 2009, KDE had reached one million source code contributions, making it one of the largest FOSS projects at the time.[8]

After the release of version 4.0, KDE changed the mission of the community to be an umbrella organisation that supports various free software initiatives, the desktop, now named "Plasma", being only one of them. In this move, KDE explicitly shifted towards emphasising the community process aspect over the development of the main product. Today, KDE is a large, still mainly volunteer-driven community with multiple regional subgroups (for example in Latin America and India). It still develops the desktop and other products, and is networked and affiliated with the Open Source Initiative, FSFE and other FOSS stakeholders.

Formal and informal organisational structure and conventions

In November 1997 KDE e.V. was registered with the mission of representing the budding KDE community in legal and financial matters. It gained charitable status in 2012. The bylaws of KDE e.V. were the first and for a long time the only written constitutional document of the KDE community. It was also for a large part copy-pasted from unknown sources and not tailored to the needs of community collaboration. When the growing numbers of contributors led to difficulties in coordination, the core contributors, instead of building an overall structure supportive for the coordination of a larger group, retired into more specialised communication channels. Almost all other behavioural norms at the time have been

[7] `https://community.kde.org` (accessed 03/02/2024)

[8] `https://dot.kde.org/2009/07/20/kde-reaches-1000000-commits-its-subversion-repository` (accessed 03/02/2024)

implicitly assumed. In August 2003, an updated version of the bylaws was accepted that for the first time was drafted specifically to support the work of the community. It introduced the concept of passive membership that enabled long-term contributors to remain members of the organisation when their level of involvement declined, without endangering voting quora, and codified the invite-only acceptance criteria for individual membership.[9] Membership in KDE e.V. is reserved for active current contributors to the project, who need to be invited to membership in the organisation by two existing members, emphasising a strong focus on individual contribution. Companies and other legal organisations cannot become full members, only (financially) supporting members without gaining a vote in the general assembly. Employees of businesses may become members, but only on their own merit and in their own right. As a result, today the community is almost exclusively driven by volunteer contributors, with businesses invited to an advisory role.[116] In August 2008, the KDE approved a code of conduct as the first documented community behavioral guideline next to the bylaws. In October 2012, the community published the "KDE Manifesto", which postulated norms like open governance, inclusivity and common ownership.[10]

The KDE community operates in a decentralised fashion with KDE e.V. as a central support organisation and offices in Berlin. Multiple regional sub-communities exist at different levels of formalisation. Some like the ones in Spain, India and Latin America are represented by individual legal entities that are associated with, but not controlled by KDE e.V.

From the beginning, KDE e.V. was meant to support and represent the community. It was clarified in the 2002 general assembly that this excluded KDE e.V. from influencing the technical direction of the community product. The organisation is commonly represented by the board, and conducts an annual general meeting. In 2005, an attempt was made to establish formally recognised working groups that would coordinate with the board and be able to manage specialised budgets. Multiple working groups were established, but most ceased activities within a few years. No other formal structure has been defined within KDE e.V. [136, 116]

From the start and into the initial stage, the KDE community con-

[9] The author of this study drafted the 2003 version of the bylaws in 2002.
[10] https://manifesto.kde.org/ (accessed 03/02/2024)

sidered itself a meritocracy. One of the first principles the initial group of about 10 contributors established was "(s)he who does the work decides", which postulates that even a community decision cannot force a technical direction on the person implementing it. The internal understanding was influenced by the publication of "The Cathedral and the Bazaar"[120], which many of the early contributors had read. The group choose a meritocratic, egalitarian approach to self-organisation, with the original founder acquiring the most impact. Communication differentiated into different channels, especially mailing lists, mainly to keep the distraction of ongoing debates away from contributors working on the product. There was general acceptance for the argument that an inner circle needs to exist to manage the project, first with the *kde-private* mailing list and later until today with the non-public KDE e.V. membership mailing list. Access to this inner circle was granted by the existing insiders.

Informal behavioural norms always played a significant role within the KDE community. There is a strong resistance to any form of authority within the community that is not based on individual merit. The rule that KDE e.V. shall represent the project, but not influence technical direction is considered a fundamental constitutional principle that newcomers would be introduced to very early on. The resistance to authority was also embodied in the "(s)he who does the work decides" norm. As a result, technical direction developed organically from the activities of the contributors. Voting and other forms of formal decision making are not highly appreciated and seen as measures of last resort. Votes are commonly conducted to accept new members into the organisation, and to elect board members and representatives to external committees or organisations. The importance of these principles contributed to the absence of organisational design in the late community stage.

One of the key debates that has never been concluded is whether or not KDE e.V. represents "the heart of the community", or is meant to be a body that complements the community without being a core part of it. The strategy of the organisation was from the beginning that active contributors should be members of the organisation, and therefore jointly own and manage funding and ownership of trademarks and other assets. To achieve this goal requires an organisation that is accepted by a meritocracy, which means to aggregate the interests of many of the core contributors. However the more influential the organisation would become, the more it would come to represent the project overall, with the board growing into a sort of project leadership. This created a conflict with the fundamental resistance to authority prevalent in the community. The community did not establish processes that are able to make decisions on questions of this constitutional sort. Decision making relied on the relevant stakeholders taking part in an extended elaboration with the goal to reach consensus. For technical decisions, this approach served the community well. The community does however not differentiate between product related technical decisions and the implementation of norms of the social process. The decision making process aimed at consensus proved to be less efficient for topics that affected all community members, where everybody is a stakeholder. Effectively, the formal organisation becomes very difficult to change, with the consensus driven process affording each individual member with a de-facto veto.

Decision making and conflict resolution

Because of the "(s)he who does the work decides" rule, a decision "manifested itself based on what ended up in the revision control system". In the early and medium stage of community growth this approach served the community well. Possible differences would be settled by arguing for the cause until an agreement was found. The decision making process relied on the organic coordination of a familiar, cohesive group with a common cultural understanding. Over the years, this attracted more contributors with the same traits, contributing to the common lack of diversity in FOSS communities: The contributors where predominantly young male software engineers. A growing community however requires increased specialisation and division of labor. Other community professions like documentation, user experience design and community management suffered from a lack of contributors as a result. Selective bias has been criticised as a hidden cost of meritocracies.[10] It can be argued that the community may not reach the full contributor potential because of it.

Participation in debates is open to all contributors, but the impact of their voice depends on their meritocratical status. This is not unusual in smaller collective action groups.[93] This attitude emphasises technical contributions over those in other fields. Minority opinions have a difficult time to get heard. Because contributions are made voluntarily, there is only little participation of specialised and minority contributors.

Conflict resolution within the community is mainly absent. Except for appealing to the board of KDE e.V. as a general fallback option, there are no defined processes to escalate a conflict with the goal to settle it. Contributors are expected to sort things out amongst themselves. In 2008 the community working group was established, together with the code of conduct, with the aim to "maintain a friendly and welcoming KDE community, thereby ensuring KDE remains a great project enjoyed by all contributors and users".[11] The community working group however only moderates using a participative approach and is not equipped with any sanctioning instruments. The only possible measure to sanction misbehaviour is a suspension of a contributor's accounts, either temporarily or, as an ultimate measure, permanently, a task performed by the system administration group. Account suspension constitutes a drastic measure, as it effectively removes the affected person from the community. It affects contributors similarly to a citizen of a country being subjected to temporary exile or revoked citizenship or a church member being excommunicated. It also strips the person so sanctioned from the means of communication needed to enable them to continue to be part of the discussion and defend their position. Account suspensions are therefore issued only in very few cases, and only after lengthy moderation did not resolve the conflict. In some cases this delayed necessary responses to disruptively abusive behaviour. Similar to the decision making processes, the mainly informal conflict resolution mechanisms work sufficiently well for product related debates with a small number of homogenous stakeholders with knowledge and a strong interest in the matter, and less well for issues with the community's social process with a large amount of stakeholder with only moderate interest. This indicates that both the decision making and the conflict resolution mechanisms have been established in the initial and medium stage of community development, and have not evolved for the late stage where the social process grows more important compared to the product aspect.

[11] https://ev.kde.org/workinggroups/cwg/ (accessed 03/02/2024)

While this description of the decision making and conflict resolution mechanisms within the community may appear as criticism, it mainly aims at describing the observable results and developments. The KDE community had good reasons to establish these processes that are founded in the ethics of the community's early contributors. Well-defined formalised decision making processes favour well-organised actors with the necessary resources to participate in them, which are expected to be businesses and politics, not volunteer contributors. The absence of formal decision making processes is seen as an emphasis of the role of the contributor on the product over others "that merely talk". Similarly, the apparent lack of conflict resolution mechanisms is by design as well. One argument is that as long as the community has difficulties defining misbehaviour, it should not police it as that would result in arbitrariness. A second argument is that contributions that are disruptive to the overall technical direction of the community are important to innovativeness and should not be suppressed.[12]

The lack of formal definition of decision making processes results in occasional over-the-top behaviour that consciously or subconsciously, sometimes with good intentions towards hearing all sides of an argument, prevent decisions from being made or from being implemented. Since there is no process that describes how discussions should be conducted, debates can become endless by bringing up a new argument or point of view that needs to be reflected. It is not easy to distinguish between the contribution of an important argument to the debate and the deliberate raising of a tangential argument with the aim of derailing or prolonging it. Bringing up tangential arguments in a debate that may or may not be critical to the conclusion happens often enough that it developed its own term, *bike-shedding*, after the question of which colour to paint the bike shed when the debate is about whether or not to build one.[51] The result is that debates may take much longer than the subject warrants, a topic will not receive the necessary attention, a decision may never be made, or a decision once made will not be implemented. Occasionally, the community applies "lazy consensus" where the debate is settled by a contributor committing a solution that reflects what has been discussed,

[12] https://community.kde.org/Akademy/2013/ConflictResolution (accessed 03/02/2024), there was no formal adoption of the suggestions.

pre-empting further discussion. This happens more often with matters where stakes are difficult to define or controversial. In the KDE community, the KOffice versus Calligra discussion[13] or the decision to hire more staff, especially an executive director, are examples for discussions that dragged on for a very long time, sometimes years, before being concluded.

This pattern is even more difficult to manage because the reasons to prolong a debate for an individual contributor may be sub-consciously self-serving, but are rationalised towards the good of the community by the individual themself, making them think they act towards the best of the community. It does give individuals an instrument to abuse a participative debate culture that generally assumes good intentions. Instruments like time-boxing (limiting the period of debate by scheduling an executive decision or vote at the end) that are common in other collective action groups are not used in the KDE community because formal decisions through votes are not generally accepted. Interviewees suspected that the debate culture in the community was heavily influenced by the student lifestyle of the early contributors, dominated by non-structured arguments, a lack of any constitutional frame of reference and a lot of time for debate.

Community membership, roles and privileges

The KDE community implements an easily accessible "open door policy" to its core repositories, defined by the absence of any formal hurdles to gain access to the community infrastructure. Since all code and data is revisioned, any change can be reverted, and there is no need for an approval process for contributor access. Only a small subset of the community infrastructure, for example the public-facing web sites, are kept under more restrictive control. Everybody who contributes to KDE products or the community ("everybody on the mailing list") is considered a community member. To participate in the *product* related aspect this is all a contributor needs. Already in the very early stages, a private mailing list for the more involved contributors was created, with an invite-only membership policy. It later evolved into the communication channel for the KDE e.V. members that is kept private until today. At the AGM in 2012, an attempt

[13] https://lwn.net/Articles/419822/ (accessed 03/02/2024)

was made to change this communication to be public. It ended in the creation of the `kde-community` mailing list[14], while the communication of the organisation is still private. There is privilege differentiation within the community regarding participation in the *social* aspect of community work, combined with significant barriers to entry like the invite-only principle. Contributors value being a community member highly, especially once becoming a part of the core team or KDE e.V.. Advancement to a role of formal community representative on the board or in external committees or foundations practically requires membership in KDE e.V. The KDE community is easily accessible for contributors to its product, but not as much to its social process.

Structural reforms and outlook

The formal organisation of the KDE community was changed only infrequently with the update of the bylaws, the introduction of the code of conduct, the publication of the manifest and the formation of the community working group, over the course of more than 20 years. Changes were incremental rather than disruptive, and retroactive in that they codified norms that the informal social process already had developed.

The informal organisation changed gradually but preserved "hacker culture". In the absence of formal structure, thought leaders have strong impact, which puts an emphasis on personality that is difficult to replace at a later time. If two contributors did not agree on which text editor KDE should ship, it would ship two text editors. There was no mechanism to influence technical decisions that affected the project as a whole and the users of the software. More importantly, there is no sanctioning mechanism to encourage activities that the community is interested in. The "(s)he who does the work decides" rule means that the user has to turn into a developer contributor to improve the software for her or his needs.

The described community norms all have been developed and adopted in the early stages of community growth. They worked well in the small to medium sized group and did not change significantly in later stages when KDE decided to de-emphasise product development over being a community that creates FOSS products. An unresolved

[14] https://mail.kde.org/mailman/listinfo/kde-community (accessed 03/02/2024)

contradiction lies in the application of predominantly informal norms and ethics tailored towards a smaller coherent social group with uniform backgrounds and interests to the governance of a large, diverse organisation. The cultural foundation that the community codified in the vision and manifest is not implemented in it's long-standing governance norms.

2.4.3 Wikimedia

Mission, foundation and history

The online encyclopedia Wikipedia was launched in 2001 with the vision to create "a world in which every single human being can freely share in the sum of all knowledge". Unlike predecessor projects, it incorporated the idea that all content should be free with the same understanding as in free software. This vision was formulated in the early days of the project by the founder Jimmy Wales, and still remains largely unchanged.
Wikipedia is created by the global Wikimedia community. There is no central authority within the community that manages activities globally. Instead, regional sub-communities operate mostly autonomously, usually along language boundaries. It is therefore difficult to describe the governance norms of Wikipedia as a whole. In terms of contributors, community activities and funding raised, Wikipedia has a strong presence in central Europe, especially the German speaking countries. This study focuses on the community of contributors to the German language Wikipedia and the supportive organisation Wikimedia Deutschland e.V. in Germany.
Initially, Wikipedia was perceived more as an idea, a broadly collaborative effort to make the knowledge of the world accessible to everybody and to enable them to participate as a user, author or community member. The idea quickly turned into a global movement that attracted a large numbers of contributors. In May 2017, about 120,000 participants actively contributed to the project. With over five million content pages and nearly 900 million edits, Wikipedia has successfully built the encyclopaedia it set out to create, surpassing commercial encyclopaedias by article count and number of readers. It gained a large user base in the process, serving in average about 7.8 billion pages per month. It globally ranks 5 in the list of most visited websites. With this initial success, some contributors shifted their focus towards building the very best encyclopaedia, with

a focus towards quality over quantity. Others identified new fields and regions of knowledge that need to be captured and consider the task of collecting all relevant human knowledge far from completed. In any case, building the Wikipedia encyclopaedia is an ongoing, life-time scale undertaking. The question whether Wikipedia's mission makes it a project responsible for social change or for the concrete task or writing the best encyclopaedia in the world is still being discussed.[15]

Formal and informal organisational structure and conventions

There is only minimal formal organisation of the community of German-speaking Wikipedia authors. Wikimedia Deutschland e.V. represents the German language Wikipedia legally and provides community support. Similarly, the San Francisco based Wikimedia Foundation legally represents the global community and the English language Wikipedia, and maintains a level of control over the regional organisations. These organisations however are not directly involved in coordinating or managing the work of Wikipedia authors and other individual contributors. Authors commonly focus on contributing knowledge in their own language and possibly to the English language Wikipedia, which is seen as the global fall-back. More than in other organisations, the contributor base is fluent, because it is possible to contribute, even anonymously, without much interaction with the organised community. Groups of regulars (*Stammtische*, in German language) meet occasionally to maintain cohesion between the work of the individual authors. Many participants expect regular contributors to attend physical meeting to gain recognition. Editorial boards have formed for specific subject matters like chemistry or religion.[16] Arbitration committees have been created in some countries (2007 in Germany) that assist in resolving conflicts between Wikipedia users. The arbitration committees do not interfere with regular contributor activity.[17] Beyond that, no formal structure exists that the authors

[15] A history of Wikipedia is available at https://en.wikipedia.org/wiki/History_of_Wikipedia (accessed 03/02/2024). Details about the vision can be found at https://wikimediafoundation.org/wiki/Vision (accessed 03/02/2024). Content page count, number of active users and other metrics are available on Wikipedia's statistics page: https://en.wikipedia.org/wiki/Special:Statistics (accessed 03/02/2024). A user is considered active if she or he performed an action in the last 30 days.

[16] https://de.wikipedia.org/wiki/Wikipedia:Redaktionen (accessed 03/02/2024)

[17] https://en.wikipedia.org/wiki/Arbitration_Committee (accessed 03/02/2024)

turn to to coordinate their work. Purposefully, no attempts are made to unify the processes of the regional sub-communities. Regional differences and decentralised self-coordination are considered key strengths of Wikipedia.

Wikimedia Deutschland e.V. develops software used by Wikipedia, especially MediaWiki, lobbies for open knowledge politically, invests into free learning and free educational resources, provides infrastructure and facilities for use by volunteers, and overall manages the organisation's and the community's legal and financial footprint. In 2016, it reached 50,000 individual supporting members, 2,000 voting members, and about 85 employed staff. It is lead by the up to nine strong executive committee (*Präsidium*), which appoints the executive director. The activities of Wikimedia Deutschland e.V. are for the most part considered orthogonal or supportive to the work of the community of authors. Some participants in the interviews actively refused the notion that Wikimedia Deutschland e.V. is part of the German language Wikipedia community and consider both separate entities. Wikimedia Deutschland e.V. does not consider itself responsible for the activities of the community of authors. The relation between Wikimedia Deutschland e.V. and Wikimedia Foundation has been characterised as that of "a far removed sovereign" that tributes are paid to.

The community organisational structure has been described as "profoundly informal". Especially early community contributors or "generally important top-dogs" can be very influential, even without formal roles. It has been pointed out in the interviews that this may impose significant barriers of entry for new authors. Formal and informal structure have deviated decisively. It can be assumed that the community is not in a position to perform an analysis of the state of the project and derive conclusions for organisational reform, which has been classified as "negligent" in the interviews.

Decision making and conflict resolution

The decision making and conflict resolution norms within the author community show strong similarities to those found in the other cases, underlining the assumption that governance norms evolve based on community composition.

Acknowledging that majority decisions cannot be enforced against volunteer contributors, they have been replaced with non-binding opinion polls (*Meinungsbilder*). All active contributors may initiate an opinion poll and participate in one. There are strong opinions about opinion polls, with some arguing that contributors should participate in them, and other arguing against participation. An aversion against formal decision making is obvious. Similarly, the "rule to ignore all rules" encourages participates to apply agency to their actions.

Intra-community conflicts are managed along a well-documented staged process from de-escalation to appeals to a mediation committee (*Vermittlerausschuss*) with volunteer members and finally to an arbitration board (*Schiedsgericht*) with members that today are elected with a qualified majority. Decisions of the arbitration board are considered binding within the community. Recommendations like remaining levelheaded and assuming good intentions help to maintain a collaborative spirit, as do more formal guides like the "Wikiquette". While there are instances of "edit wars" or members acting under fake accounts ("sock puppets"), the conflict resolution process is mostly accepted and effective. These processes represent a mature understanding of the role of decision making, conflict resolution and of volunteer community dynamics. None of them involve Wikimedia Deutschland e.V..

Community membership, roles and privileges

Everybody who productively contributes to Wikipedia is considered a community member. Since anonymous contributions are allowed, contributors transition from loosely associated anonymous authors to registered authors known by a screen name and then may acquire additional roles like administrators. Elected *Bürokraten* (bureaucrats) manage administrator status. A number of additional roles exist that partially map to technical permissions in the operation of Wikipedia, like rolling back changes or inspecting contribution metadata. There is consensus that all contributors should be considered equals, taken seriously and valued based on merit. Even though being admin is foremost a technical task that allows to change other contributors' content, it is also implicitly a social role that needs backing by the community and therefore a strong standing or merit for the person acting as admin. Eligibility to vote is based on a minimum number of recent contributions, and since the bar is set rather low, has become a requirement for effective participation in discussions. Without it, an individual "would not be taken seriously". Long-term contributors that shifted their focus towards other activities than being authors sometimes struggle with that or produce edits to maintain their status. Social status within the community is closely related to contributions either or quality content or to the software used to run Wikipedia. Contributors to auxiliary functions like conference organisation or design are "not well known".

These status groups or roles represent contributor functions with a strong product focus - they are measured against their impact on the quality of the encyclopaedia. A remarkable disconnect was mentioned in the interviews between the legal entity Wikimedia Deutschland e.V. representing the German speaking Wikipedia, and the community of authors. One described the role of Wikimedia Deutschland e.V. as "collecting donations, being on TV, and attending galas, based on the work of the community". Multiple interviewees mentioned that being a member of Wikimedia Deutschland e.V. was perceived in the past as a negative factor with regard to contributor merit within the community, and is now considered "acceptably eccentric". The role of Wikimedia Deutschland e.V. members within the community is largely irrelevant, except for a small number of contributors that try to participate in both organisations, but find it time consuming and difficult. Wikimedia Deutschland e.V. has been repeatedly criticised for being disengaged from the community and not supporting it enough. Interviewees expressed that they believe the perceived under-performance of Wikimedia Deutschland e.V. is rooted in the lacking integration with the community, and that they "are happy if Wikimedia Deutschland e.V. is at least not breaking anything". Wikimedia Deutschland e.V. keeps authority of the budget and spends a significant share of the budget on non-product related activities.

Structural reforms and outlook

Similarly to the other case studies, the formal organisation within the German Wikimedia author community has rarely changed. There was no structured process of organisational design review. In the past ten years, the adoption of the review principle, the arbitration court and the introduction of the visual editor are perceived as the major changes. Interviewees described the overall constitution of the community as rather “hostile to change”.

Wikimedia Deutschland e.V. is aware of the rift between the author community and the formal organisation. It attempts to integrate the community through a collaborative planning and budgeting process and other activities. Decreasing author numbers create the necessity to act upon a perceived pent-up need for organisational reform, which is reflected in the annual plans for 2016, 2017 and 2018. Attracting and retaining volunteer contributors has been accepted as one of three key fields of action. However, less than ten percent of the overall revenue from donations and membership fees is allocated directly towards that goal. Wikimedia Deutschland e.V. positions itself as an organisation with the primary goal to foster Wikimedia projects.[18] There is a profound feeling within the community of authors that Wikimedia Deutschland e.V. made itself independent and unaccountable. When asked what would need to change, one interviewee suggested that community members “get together” and re-take control of their project.

2.5 Observations

For the most part, the contributors agree on what they expect from their communities: They want to engage in a community of makers. Amongst their peers, they wish to have equal opportunity to contribute. They understand the need for community management, but want their communities to remain focused on being ambitious, productive meritocracies. They believe that there is strong solidarity between the members of their communities, and that “them-versus-us” conflicts between the communities and their leadership or the makers and the community builders are mainly absent. Still, the communities are exhibiting similar symptoms of distress: They have trouble growing their contributor and contributions

[18]“Wikimedia Deutschland [ist] im Hinblick auf die Wikimedia-Projekte daher als "Förderverein" zu verstehen.” (Wikimedia Deutschland Präsidiumshandbuch (accessed 03/02/2024))

count sustainably, have difficulties implementing organisational change and get stuck making important decisions, resolving inner-community conflicts or enforcing the values of their social groups.

It is obvious that the contributors think highly of their communities. There have been no indications to assume any malicious intend by influential participants to abuse the communities for their own advantage. Conflicts within late stage FOSS communities are more likely to reflect difficulties volunteer contributors have to collaboratively develop their organisations and maintain control over their destiny as they grow to be large groups. The similarities of the symptoms between the communities illustrate this observation.

2.5.1 Formal and informal organisational structure and conventions

Formal organisation is not the first thing participants have in mind when starting a FOSS initiative. The groups are small initially and do not possess assets or liabilities that require an independent formal organisation. KDE and Wikimedia Deutschland e.V. added a legal entity that represents their community later after their projects started. The founders of FSFE on the other hand where aware that their success will depend on a strong, independent organisation, and started off with a carefully designed organisation that anticipated attempts of hostile take-overs and the creation of regional subsidiaries.

None of the three organisations implemented a systematic effort to periodically review and reform their formal organisation. Changes to the organisational structure have been very rare, resulting in a growing disconnect between the community's production processes and their governance related activities. FSFE did not succeed in establishing thriving, decentralised, independent regional sub-organisations, and concentrated its activities at the Berlin head office. The KDE community continuously restricted the mission of KDE e.V. to administrative support, and resisted the delegation of authority to elected representatives. This led to long-standing contributors questioning its usefulness, and contributed to an emerging culture of bike-shedding and indecision. The organisational structure of KDE e.V. was not changed even as the KDE community changed from a single-product to an umbrella community. Eventually, KDE e.V.'s main role became to organise the annual KDE Akademy[19] conference and to provide funding for contributor meetings. For a period of time, the KDE community became infamously known for its lack of coherence and decision making. However, the strongest rift between product development and formal organisation in this study is exhibited by the German language Wikipedia community. Where besides minor differences FSFE and KDE still see their formal organisations as an integral part of the community, some members of the community of Wikipedia authors wish that Wikimedia Deutschland e.V. "not interfere with their work", while being a Wikipedia author is explicitly not considered a selection criterion for employment at Wikimedia Deutschland e.V. The volunteer community of authors and the formal organisation that bears the community's name have diverged. The performance of the support organisations in this study is not linked to the effort invested into the original organisational design. Instead, given the absence of a systematic review and reform process the ability of the organisations to serve their communities deteriorated as they went from the initial through the medium to the late stage. The aversion of the contributors against authority and "bureaucracy" puts the need for reform in question and reinforces this trend. It is not the initial design that counts, the organisations need to continuously adapt and improve the performance at which they support their communities.

All three organisations rely heavily on informal organisational structure. There is strong agreement between the interviewees that the documented formal structure is not implemented in reality, and that today the organisational structure of the communities is mainly implicit, well-understood only by early community members, and not well-documented for newcomers. Only few fully know and understand the existing formal rules as they stand today. The divergence of formal and informal organisation and the lack of supportive performance of the community organisations is not perceived as a currently relevant problem. The resulting effects, enforced by lack of positive competitive selection of community leadership or inhibited acquisition of new contributors, are detrimental to long-term community growth and success.

[19] `https://akademy.kde.org` (accessed 03/02/2024)

2.5.2 Decision making and conflict resolution

It is commonly part of the spirit of a FOSS community that decisions should be made by consensus, that authority and hierarchy should be avoided, and that there should be minimal to no policing of contributor activity. These are all positive, defining aspects that are important to contributors. But do they match reality when compared to the decision making processes and conflict resolution mechanisms of late stage communities?

The results from the interviews strongly suggest that all three communities make use of very few defined decision making processes, do not routinely apply instruments for shaping debates, experience extended bike-shedding and indecision regarding issues that are considered important, and that influential individuals, often project founders or early contributors, wield soft and hard vetoes over community decisions.

Most day-to-day decisions are made at the level of subgroups that focus on particular aspects of the community product. In these relatively small groups, informal decision making still succeeds. It is possible to understand the likely outcomes of the decision, and there is a joint sense of responsibility for that result. There is also no need for an appeal mechanism. If the outcome of the decision is not what was expected, the group again jointly decides on a new course of action. These are the decision making mechanisms the communities developed in the early stages and that served them well.

Late stage communities also need to make more complex decisions, like hiring an executive director, organising a global conference or redefining the overall community vision and mission. These may involve trade-offs of resource allocation between subgroups or competing goals. The community as a whole is a stakeholder in these decisions. The outcomes of the decisions may be harder to predict, and unlike most technical decisions difficult to reverse. Undefined and informal decision making processes, a lack of routes of appeals, and an excessive debate culture that may prevent decisions from being made have a detrimental effect on contributor motivation, and pose a significant barrier of entry into higher level community functions. Early contributors stay in community leadership roles overly long, at the expense of later contributors not assuming leadership roles even if their merit within the community would warrant it. The auto-organised decision making mechanism of the subgroups fails when applied to higher level large group decisions.

Authority is commonly assigned to specific community functions, like the president in the case of FSFE, or the board in KDE e.V. There are no checks and balances however to decisions made by these functions. Even if it may be known to some participants that a way to question a decision of the president is to submit an item to the agenda of the next general assembly, this is far from obvious to the wider community, and also not communicated. There is little understanding that for every community function with authority a check needs to be implemented that implements oversight and allows for decisions to be appealed. In volunteer-driven communities with self-referential authority, this means that otherwise unresolvable issues eventually will escalate to a community all-hands decision. In turn, this requires a mechanism for community votes. The aversion communities have against majority decisions is well-founded, especially considering that all contributors participate in the communities voluntarily. However as a mechanism of last resort, no better alternative has been presented. Every decision should be appealable. Many of the long-term contributors interviewed in this study today acknowledge the need for well-defined decision making processes, even though it took a long time for them to to reflect on and change their initial preference to auto-organisation at all levels.

With unclear authority, it is difficult to apply instruments to shape debates so that a decision can be achieved in a reasonable time frame. Discussion drag on "until nobody has any energy left to disagree". A "fulsome optimism regarding the decisiveness of a large group" can be observed. Discussions are being kept alive by influential contributors to avoid their conclusion with an unwanted decision. Shaping debates does not necessarily require voting mechanisms. Time boxing by asking that a consensus be reached after a specified discussion period and announcing a formal vote otherwise is one option. Relying on rough consensus combined with a clear procedure of appeal is another one. The KDE community made good experiences with a "debate manager", a contributor that voluntarily steps up as a moderator and drives the debate to a conclusion. It can be expected that by combining decision making and appeal processes more clearly and organising debates in a more result-oriented fashion, the tendencies towards bike-shedding and indecision that communities exhibit can be overcome.

2.5.3 Community membership, roles and privileges

The communities apply a broad definition of what makes a community member. “Everybody on the mailing list” who actively participates is considered to be one. Those who contribute more over an extended period of time begin to form a loosely defined “core team” early in the process, which also separates those who “merely talk” from those “doing the work”. Formal membership in the support organisation forms another community rank. Being appointed to a board or elected leadership position is another one. This suggests a hierarchy of influence that may be misleading, as advancing through the community ranks does not necessarily happen on a straight career path. The differentiation between product contributions and community management may lead to contributors gaining leadership positions that never contributed to the community product. Authority is also gained ad-hoc by individual contributors self-identifying with the initiative to manage a debate, or a community process like writing the manifesto. It can be observed that once contributors reach a board or elected representative level position, they rarely ever go back to being regular contributors. This indicates that such positions do form a sort of end-of-career achievement. Community rank is considered significant in that individuals would, for example, list their community achievements in their CV.

Contributors advance through the community rank meritocracy based on their contributions. Not all contributions are valued the same. Contributors to the core product, founders and individuals “rich on time” advance through the meritocracy more easily. Typically, contributors gain more merit when contributing directly to the community product, as opposed to covering support functions as in helping with administration or event management. Even auxiliary product contributions like the work done by designers and documentation authors have more difficulty in achieving appreciation. This may inhibit effective specialisation as the different “professions” within the community carry different merit. The inherent contradiction of the "who does the work decides" rule applied by many communities is that in an advanced community it is almost impossible to identify which specialised task is more important and who does the work.

The phase during which the contributor joins also affects the opportunities to advance through the meritocracy. Project founders and early contributors often remain in an influential position over a long time. Sometimes they evolve to be "luminaries" or "top dogs" that carry strong influence over community processes and are involved in many community decisions even without a formal role. Long-standing and early contributors aggregated merit that enables them to influence the community as a whole. This poses a difficult barrier for newcomers to become contributors, and even more for existing contributors to advance within the community status groups. The communities are aware of barriers of entry and work to keep them low, it seems however that the barriers are higher, not lower, for advancement and individual personal development *within* the community.

Many of the interviewed project founders and early contributors that rose to community leadership functions in the early and medium stages of the project emphasise that their motivation was to help the project, not to further their personal reputation. Some say they where willing to do the work no other contributor wanted to do. Others stress that the perception of the president's position is of much higher value to others than to them. Their expectation towards other community leaders is that they would also mainly work towards the interests of the community, not their own advantage. Some interviewees admitted that other contributors may regard the group of founders and early contributors as a "round table" that is a bit out of touch with the rest of the community. The modesty expressed by the founders and early community leaders is convincing in the early and medium stages of community growth. For late stage communities, it must be assumed that the prestige and also remuneration for serving in a community leadership role becomes an attractor in itself. Late stage communities will then require a system of checks and balances to maintain control over community management, which was unnecessary and therefore not established in the early and medium stages.

Similar to organisational structure there is an implicitness in the community status groups, roles and privileges. The self-referential authority within the community is well-understood by the founders and early contributors. One interviewee said “I set the rules once, I can do it again.” This freedom to question rules and apply norms where they are applicable and ignore or bend them otherwise is second nature to old-timers. It is explicitly communicated, as in “if a rule prevents you from improving or maintaining Wikipedia, ignore it”, but difficult to grasp for newcomers. Rules solidify, sometimes unwantedly, and are rarely ever changed for a late stage community. The open doors policy that the communities are proud of deteriorates in the late stage. Long-standing administrators expressed worries that “some of use have lost the trust that newcomers will do good things”.

2.5.4 Structural reforms and organisational design

In all three cases, the explicit and implicit organisational structure emerged in the early and mid stages of community development. While the initial structure developed implicitly, the communities did create well-working formal supportive organisations that originally served their purpose well. They did however not implement a systematic and periodic review process which ensures that implicit and explicit structure and processes do not diverge too much, and that the formal organisation stays focused on the mission that the community created it for. They also implemented partially insufficient checks and balances to enforce accountability of these organisations towards their contributor base. Through membership open to all active contributors and direct as well as competitive election of community representatives by the members, KDE e.V. remained most effective and accountable to the community of the three cases. With the removal of the elected fellowship representatives and the position of the executive director, FSFE grew less accountable in 2018. While it does still represent the ideals of software freedom and aims to speak for the wider open source community, it gains few new contributors. Of the three cases, Wikimedia Deutschland e.V. developed to be most removed from its original purpose of serving the German language Wikipedia community. While it drives fundraisers and shares the name of the community project, most of its activities and most of its budget spent do not directly support the community of authors. While all its activities are charitable and contribute to the cause of free knowledge, almost all they share with the community of authors is the name. It is reasonable to assume that this aspect contributes to the declining number of authors as potential contributors realise that their work is being used to raise funds for mostly unrelated activities and a large body of staff.

All communities exhibit an aversion against administrative processes or "bureaucracy", resulting in an apparent lack of impulse towards active organisational change. Interviewees from all three communities mentioned that formalising structure and documenting decision making and conflict resolution processes to a necessary extend does help maintain the freedoms to participate and joy from contribution. They face the challenge of preserving "hacker culture" while at the same time enabling large numbers of contributors to collaborate successfully. Based on the lack of organisational design, community processes and structure are largely implicit and there are no well-defined rules of appeal. The German-speaking community of Wikipedia authors provides a positive example of well-working auto-organisation. However these processes are independent of Wikimedia Deutschland e.V. as their support organisation, indicating that the resulting lack of accountability offers opportunities for self-serving behaviour.

This raises the question of how communities can ensure that their structure and processes evolve so that they continue to fulfil their mission of supporting the contributors. Where competition keeps businesses aligned with their purpose and elections align the actions of politicians with the interests of the population, FOSS communities depend on voluntary participation to raise contributions. This postulates the number of independent contributing entities and the number of contributions raised as key metrics for community health. Implicit and explicit community structure and processes should primarily support these goals. Community activities, also by the support organisations, should be assessed based on how they contribute to these goals. From the budget a community is raising, every Euro that is spent on activities that do not contribute to these goals reduces the number of attracted contributors and through that the potential impact and success of the community. The self-referential purpose of FOSS communities means that all functions of the community need to be accountable to the base of its active contributors. In turn, a community can only represent those that by way of actively contributing acquire an equal voice in decision making and conflict resolution processes.

2.6 Summary

This study started out from the observation that FOSS communities struggle to maintain growth once they reach a large number of contributors. It could be observed that the growth phases the communities proceed through can be grouped into an initial stage with ad-hoc coordination and an equivalence of individual and group goals, a medium stage with consensus-focused auto-organisation and a late stage with more profound functional differentiation and formal structure.

Businesses, individual volunteers and staff members participate for different sets of reasons. The concept of community composition refers to the mix of volunteers, businesses and staff that engage in a community. Assuming that, all else unchanged, governance norms develop depending on community composition, the study analysed three primarily volunteer driven communities to provide an insight view of their governance and to identify commonalities between them even though the create vastly different products. Based on the principle of voluntary participation, the purpose of communities is defined in a self-referential manner: The community serves the interests of the contributors that form it, with no outside authority except the law. This means communities need to solve the constitution problem to define who has a voice and to establish structure as well as decision making and conflict resolution processes, based on voluntary participation.

Community governance is shaped by the mindset of their contributors. Individual volunteers are primarily intrinsically motivated, which is reflected in the expectations they expressed in the interviews: to participate in a community of makers, to experience equality of opportunity among their peers, to find a balance between makers and community builders, to become a part of an ambitious, productive meritocracy and to see their own ethical principles represented in the community governance norms. If these expectations are fulfilled, they develop increasing loyalty towards the community.

The case studies reflected the concepts of membership, the formal and informal organisational structure and the decision making and conflict resolution processes of the communities against these expectations. They indicate that while there is a close or close enough match in the initial and intermediate stages, especially the formal organisations that have been created to support the work of the community show a tendency of distancing themselves from the community goals. The combination of solidified implicit norms and more closed-up organisations creates barriers to entry for newcomers and reduces the number of long-term, loyal contributors the community is able to attract in the late stage. While it remains relatively easy to contribute to the community product, it becomes increasingly hard to gain access to influential formal roles and positions.

The gap between makers and community builders grows with size of the community. Independent of the effort invested in setting up the original support structure, the formal organisations partly disconnected from their communities. This seems to be caused by the absence of a regular review process based on checks and balances built into community governance, resulting in a lack of accountability of the support organisations towards their constituencies. Volunteer contributors exhibit aversion to authority and formal decision making. At the same time, they jointly are the highest authority within their community. A possible conclusion is that community decision making processes should be well-defined, and that the highest level of escalation should be the community as a whole. Conflict resolution mechanisms should mirror the decision making processes.

The communities investigated in this study partially lack instruments to ensure that their structure and processes support the overall community goals. Similar to elections in politics and supervisory boards representing investor interests in enterprises, the communities will need to re-align decision making power and accountability to remain volunteer driven in a successful transition into the late stage of community growth.

Chapter 3

Open source software in standard setting: The role of intellectual property right regimes

This article was originally published as: Knut Blind, Mirko Böhm, and Nikolaus Thumm. “Open Source Software in Standard Setting: The Role of Intellectual Property Right Regimes”. In: *Open Source Law, Policy and Practice*. Oxford University Press, Oct. 2022 (pages 256-272). Reproduced by permission of Oxford University Press. For permission to re-use this material, please visit `https://global.oup.com/academic/rights`.

This chapter considers the intersection of FOSS and FRAND licensing and its integration into the process of standard setting. FRAND commitments aim to prevent IPR holders from refusing to license patents, and from charging licensees excessive fees (unfair or unreasonable) for standard implemented patented technologies. The complex interface of FOSS and standardisation processes is analysed with a specific focus on the role of IPR including FRAND licensing. Standards and open source development are both processes widely adopted in the ICT industry to develop innovative technologies and drive their adoption in the market. Innovators and policy makers often assume that a closer collaboration between standards and FOSS development would be mutually beneficial. The interaction between the two is however not yet fully understood.

3.1 Introduction

In the communication "Setting out the EU Approach to Standard Essential Patents" the European Commission announced in 2017 that it would analyse complementary possibilities for interaction and differences between FOSS and standardisation processes, and recommend solutions for the smooth cooperation between standardisation and FOSS.[46] Prior to this, the interface between FOSS and standardisation had been only marginally touched both by researchers and practitioners in standardisation bodies or FOSS communities.

This chapter builds on results of the European Commission report by Blind and Boehm on the interrelation between standardisation and FOSS, which refers in particular to the interaction between FOSS and FRAND patent licences in standardisation.[15] FRAND commitments aim to ensure that essential technology protected by IPR included in a standard is made available to users of that standard on fair, reasonable, and non-discriminatory terms. FRAND commitments aim to prevent intellectual property (IP) holders from refusing to license patents and from charging licensees excessive fees (unfair or unreasonable) for standard implemented patented technologies.

The intersection of FOSS and FRAND licensing and its integration into the process of standard setting is considered throughout this chapter. It does not consider other FRAND licensing related issues, nor does it analyse specific FRAND court decisions. Comino, Manenti, and Thumm provide an overview of the wider FRAND-related economic issues,[33] and Pentheroudakis, Baron, and Thumm provide a comprehensive analysis of the most important FRAND court cases.[112]

The complex interface of FOSS and standardisation processes is analysed with a specific focus on the role of IP including FRAND licensing. Standards and FOSS development are both processes widely adopted in the ICT industry to develop innovative technologies and drive their adoption in the market. Innovators and policy makers often assume that a closer collaboration between standards and open source development would be mutually beneficial. The interaction between the two is however not yet fully understood, in particular with regards to how the IP regimes applied by these organisations influence their ability and motivation to cooperate. Most SDOs use FRAND licensing terms, while widely used FOSS licences like the GPL are largely incompatible with these terms.[94]

3.2 Results from the literature

Relevant studies are divided into three categories, according to Lundell and Gamalielsson [90], and Clark [31], without immediately taking into account the tension between FOSS licences and the FRAND regime regarding patents:

- The first cases begin as standardisation projects within formal or informal standardisation bodies that are eventually implemented as FOSS projects.
- In the second scenario, the software is initially implementated as a FOSS project, followed by a subsequent standardisation process.
- The third and last option is the parallel development of standards and their implementation as FOSS at the same time.

3.2.1 Standards initially implemented as open source software

Examples of the implementation of standards via FOSS are discussed but are mainly developed in SDOs under a RF licensing scheme, for example at the Organization for the Advancement of Structured Information Standards (OASIS) and the World Wide Web Consortium (W3C). According to Phipps, these SDOs are characterised by an implementation-oriented rather than requirement-oriented approach to standardisation.[115] There are also some standards published by the SDOs using the FRAND regime. However, there are no declarations of SEPs referring to these standards.

Nevertheless, due to the general contradiction between the FRAND regime and some FOSS licences, there is still a latent fear of conflicts with potential SEP holders, and the popular GPL licence is not compatible with FRAND.

There are some assertions that FOSS licences, such as the MIT License (MIT) or BSD License (BSD), may be compatible with FRAND.[74] However, there is no general consensus on this conclusion, as others argue equally that these are only complementary licences and are not compatible.[48, 9] Furthermore, the idea of incompatibility of specific licence systems with a FRAND regime is supported by a significant percentage of Open Source programmers.[90]

The concerns of the FOSS communities regarding the lack of clarity of FRAND licensing conditions is becoming more important due to the increasing relevance of the successful implementation of standards for quality and success. These may also, as with other patent concerns, be countered in a number of defensive IP structures, as discussed by Bain and Smith.[6]

3.2.2 Open source software as input into a standard

The second category, according to Lundell and Gamalielsson,[90] is characterised by the initial implementation of open source software, which ultimately leads to technical specifications of standards, which Phipps also calls "implementation-led standardisation".[115] In this scenario, a software implementation precedes the development and approval of the technical specifications of a standard published by either a formal or informal SDO. According to Li, it is generally more complicated for SDOs to use FOSS working practices to develop standards.[86]

In addition to the inclusion of software code in the technical specifications of standards, the functions of the code can also be transferred to a standard. Li makes a distinction here between the different licences applicable to the FOSS code. If the licence does not contain a patent clause, the patent issue is still important under the policy of the relevant SDO, possibly subject to a FRAND obligation (under Li).[86] However, if a licence contains a patent clause, the patent right in that licence is granted on an RF basis and leaves open the question of whether SDOs can require patent holders who contribute patents to the standard to license these under FRAND licensing, where there is already a FOSS licence with RF patent licences, and how any inherent conflict between the two would be resolved. Li states that this is not yet established in the current IP regulations of the SDOs.[86]

Since most SDOs have not incorporated specific rules for licensing FOSS code into the specifications of standards, Li concludes that the FOSS licensing terms are the 'only clearly applicable rule'.[86] Consequently, the granting of RF licences for the use of the code in embedded standard technologies should be applied.

However, such rules could be a strong disincentive for at least some innovators holding patents, as this would remove the possibility of charging royalties on SEPs, which might be one of the main incentives for many innovators to contribute to standardisation (see for example Lerner and Tirole [82]). In the survey by Blind and colleagues, patent-owning companies rated the relevance of the freedom of action achieved by a standard as a much higher incentive than any associated royalty payment.[18]

In addition to the conflict between the licensing conditions for FOSS and patents, there is a systemic conflict in licensing software under the FOSS licensing conditions and patents under FRAND. FOSS licences follow a cascade effect that restricts implementers in other areas not covered by FRAND.[86] Although patents are used free of charge, licences generally contain a 'patent retaliation clause' that prevents recipients from litigating against the work, including the patented contribution by terminating the patent right. This is intended to prevent implementers from filing a lawful lawsuit if they find that their patents contained in the same work have been infringed. However, the current IPR regimes of the SDOs guarantee patent holders this possibility.

In summary, the current frameworks utilised by both formal SDOs and informal consortia seem to allow the integration of FOSS into their standard development process and standards. SDOs, such as W3C and OASIS, have more of an RF culture with regard to patents and consequently have a rather limited number or no SEPs at all, and are pioneers in launching proactive initiatives to include FOSS in their standards. Despite the challenges for SDOs using FRAND, the exclusion of FOSS code from the specifications of the standards is not a sustainable strategy, as the available code base is already large and widely adopted which continues to grow.

In addition, some FOSS communities claim to set de facto standards, which calls into question both formal SDOs and informal consortia.[148]

Finally, both the increasing competition between SDOs and consortia and the additional competition from the FOSS communities as additional standard setters are likely to increase the pressure to cooperate with the latter. Industry standards can be developed through competition between communities, with no formal specification at all.

3.2.3 Open source software and standardisation in parallel

In the two previous categories, a clear distinction was made between the starting point of the process and the transfer to the other area. Category three, however, represents the interaction between the development of technical specifications of a standard together with the development of one (or more) implementation(s) of technical specifications of a standard in FOSS.

Lundell and Gamalielsson [90], who further develop Gamalielsson et al. [56], analyse the bi-directional influences between the FOSS project Drupal, which is distributed under the copyleft GNU General Public License v2.0 (GPL-2.0) licence, and the development of the Resource Description Framework in Attributes (RDFa) standard for data exchange on the web at the W3C. Support for Resource Description Framework in Attributes 1.0 (RDFa-1.0) was achieved in Drupal through its first implementation in the core of Drupal 7 (RDFa is implemented in a separate module in Drupal).[90]

The summary of the findings in connection with the third category of parallel developments in FOSS and standardisation confirms the observations made in connection with the first and second categories. In the early days of the Internet, the Internet Engineering Task Force (IETF), as a consortium driven by individual members, like FOSS projects, was involved in the development of an email format in parallel with FOSS projects. The few cases of close interaction between FOSS and standardisation are mainly concentrated in consortia with a strict RF and rather patent-incompatible licensing policy, that is W3C or OASIS. It has already been noted that they are in a better position to integrate input from FOSS projects, as opposed to the formal SDOs that use the FRAND structure.

The recursive integration of inputs from standardisation or FOSS can lead to a ‘virtuous circle’ of standards of higher quality and wider dissemination. In contrast, the challenges for the FRAND-based SDOs and consortia will also create difficulties for parallel innovation developments. In the long term, higher quality standards due to FOSS inputs combined with their wider dissemination through FOSS adoption will further increase the pressure on formal SDOs and informal consortia under the FRAND regime.

The limited focus of SDOs on copyright in general and on software or FOSS in particular has a number of economic implications. First, the few SDOs or consortia that deal explicitly with software are able to develop a stronger profile in the standardisation of topics based on software alone or on the combination of software and hardware. Here, the actors with a need for standardisation obviously decide according to the perceived competencies of SDOs, including governance in connection with software. Secondly, the FRAND regime, which is relevant for the licensing of SEPs established in traditional SDOs, in other words members of ISO, and therefore guided by ISO/International Electrotechnical Commission (IEC)/International Telecommunication Union (ITU) policies on IP, does not necessarily attract FOSS contributors used to more RF-dominated licensing systems.[8] Therefore, separation or division of labour is likely to continue in the future, despite the considerable efforts of European Telecommunications Standards Institute (ETSI), in particular, to find solutions for the coexistence of FRAND and Open Source licences, as expressed by some of its members in the Fair Standards Alliance.[2] Secondly, the rather strict RF-based policies of OASIS and W3C, which follow the Open Source licensing regime, facilitate the implementation of their standards.[8] Thirdly, the IP policy of SDOs in relation to software is linked to their business models, especially those that do not make their services freely available. It is more difficult to sell standards under a freely licensed regime that integrates FOSS into standards. However, this area of tension has not yet been addressed in the rules of the SDOs and consortia.

3.2.4 Summary of the literature

The background, as presented in the literature review, leads to very general conclusions regarding the growing importance of standard-setting processes and their use of IP, not only patents but also copyright to software. The general assessment is that standardisation plays a multidimensional role; it mediates between science- and technology-driven research and innovation and demand-oriented innovation policy, which is framed by various regulatory regimes.[103] Patents and software, including FOSS, are the main IPR used as input for ICT standardisation and are also relevant for the accessibility of ICT standardisation results. Therefore, the use of IP in standardisation processes adds another dimension.

In general, standards are developed by a number of different actors in a voluntary, consensus-based process. In view of the associated increasing diversity of interests of the actors involved in standard-setting, governance in standard-setting determines the success of SDOs in terms of integrating the various interests, which is also called for by the European Commission (EC). Effective rule-setting and governance of SDOs are critical to the successful development and, ultimately, implementation of standards. IP policies developed by SDOs will have to take into account not only specific rules and procedures for FRAND licensing relevant to patents but even more so the treatment of FOSS.

3.3 Insights from case studies and stakeholder consultation

The recent study by Blind and Boehm for the EC comprises a detailed empirical investigation of the interaction between standard development organisations and FOSS communities.[15] It is based on twenty case studies, a survey of stakeholders (more than 300 respondents) from SDOs and FOSS communities and an expert workshop. The case study analysis revealed different views on the nature of possible collaboration between SDOs and the wider open source community. The most commonly used frame of reference for 'working with the open source community' within SDOs is the expectation that SDOs will develop specifications into standards, and the FOSS community will then implement them. This approach is based on the assumption that a specification is initially created as part of a standards development process, and the creation of a concrete, compliant product is left to the implementers competing in the market. As discussed, this specification-oriented approach to standardisation is only used in a minority of Open Source instances.

The cases and literature revealed that two-thirds of these can be considered as highly innovative, large-scale collaborations that have found broad or sometimes worldwide acceptance, for example Java, Linux, and PDF. Almost one-third had a significant impact on a specific market segment. In the most recent cases, no impact has been realised as of yet, although they are considered innovative by the participants. About half of the cases achieved market-wide relevance across several industries as basic technologies or by promoting business-critical infrastructure.

There is no clear definition of industrial sectors and sub-sectors concerned, as technologies dealt with in the case studies have many uses. As a general cross-industry trend, computer and telecommunications systems are becoming fundamental technologies for various products and business processes.

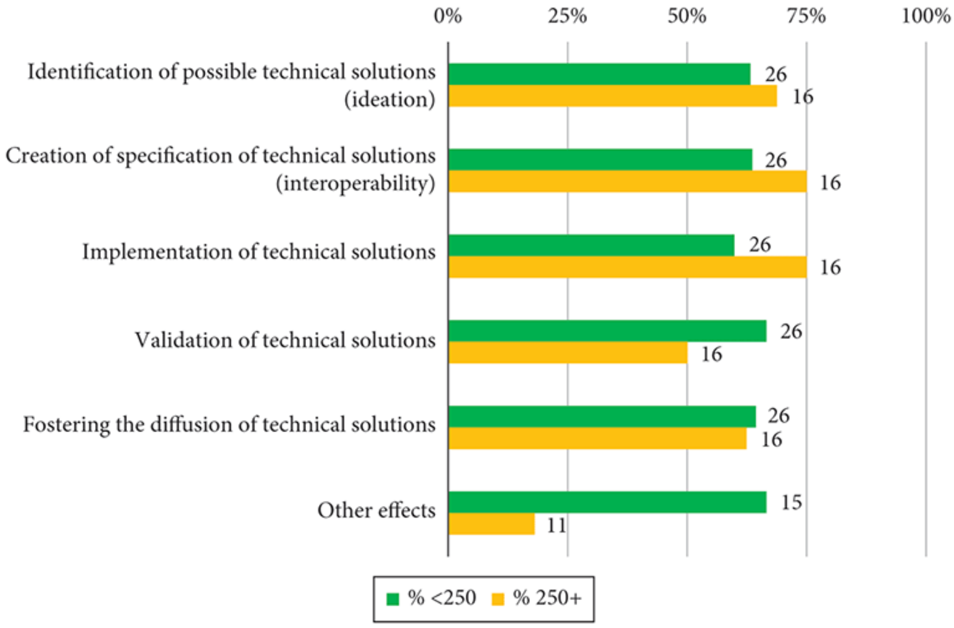
The choice of an early, parallel, or late approach to standardisation neither limits the chances of success of a project nor is a specific approach as a prerequisite for a successful standard. However, the incubation of new technologies and functions today is more often done through joint implementations or reference implementations under open source licences. Most of the innovations to which the cases refer are presented to the SDOs as soon as proven implementations exist and are generally available. Some participants in the case studies stressed that they do not see the development of standards as a means of creating real innovation but as a means of building industry consensus on available technologies to enable economies of scale. In general, models of governance and cooperation must be seen by the relevant actors as appropriate to motivate them to participate, since the most widespread technologies are also those that attract a large number of participants in their development.

In summary, the case studies have developed a partial focus on the networks and telecommunications sub-sectors. This was expected, since the interaction between standards and development is naturally located at software–hardware interfaces. The telecommunications industry has a more established history of standards practices. The cases illustrate that there are numerous successful collaborations between standards and open source development and that they have developed mature, well-established governance, such as Ecma (ECMA) TC39 or ISO JTC1.[8] Both FOSS and SDO procedures are suitable for the development of technical solutions on both a small and large scale. Those of the observed collaborations that introduced explicit patent licensing systems opted for RF ex ante licensing with symmetrical terms between contributors and between domestic and foreign licences. SDOs are usually not the driving force for technical developments. More often, FOSS communities hatch new technical solutions until they become candidates for standardisation and market penetration. The FOSS umbrella organizations and foundations (see Sandler [128]) increasingly offer functions such as platforms for collaboration and consensus building, which have traditionally been provided by SDOs.

In the study by Blind and Boehm [15], stakeholders were also asked about their assessment of the interaction between FOSS and standardisation in terms of efficiency and results. The majority of participants saw a positive effect of this link. In particular, around 70 per cent of respondents saw a positive impact on the development of specifications for technical solutions contributing to interoperability and on the implementation of technical solutions. The benefit of standardisation lies less in the idea of new technical solutions and more in their validation and ultimately in their dissemination, since only about 60 per cent of those surveyed expect positive effects in this area. Negative effects of the interaction on standardisation are generally not to be expected. The distinction between small and large organisations shows that the former tend to expect a positive impact on the identification of possible technical solutions, that is the idea finding, and the drafting of specifications of technical solutions, that is interoperability, while the latter see the benefits particularly in the implementation of technical solutions.

Looking at the effects of interaction on open source, we observe an even higher proportion of respondents perceiving positive effects. More than 75 per cent expect a positive impact on the development of specifications for technical solutions, particularly in the context of interoperability, and on the implementation of technical solutions. While about 70 per cent see positive effects on FOSS both for the identification of possible technical solutions and their dissemination, less than 60 per cent of the respondents expect positive impulses for the validation of technical solutions. Again, no negative effects were found. The distinction between small and large organisations, in contrast to the expected impact on stan-

Figure 3.1: Positive impact of interconnection of Open Source and standardisation on efficiency and results of Open Source—SMO vs LO [15, p. 156]



dardisation, shows that the former expect rather positive effects on the validation and dissemination of technical solutions while larger organisations again see the advantages in the implementation of technical solutions, but also in the identification of possible technical solutions in the field of open source.

Comparing all assessments of networking in terms of efficiency and results, it becomes clear that smaller organisations perceive the knowledge flow from FOSS to SDOs in such a way that the latter receive new ideas as input for technical solutions. Larger organisations see advantages for SDOs in implementing technical solutions as FOSS. In contrast, smaller organisations experience positive effects of standardisation on FOSS on the validation and dissemination of technical solutions. There are a complementary impacts explained by the size of the organisations.

3.4 Compatibility of intellectual propery regimes in SDO and FOSS

The limited research that deals explicitly with the interaction of SEP licences and Open Source focuses primarily on the legal compatibility of FOSS licences with the FRAND licensing of SEP. This is an important dimension of interaction, since any directly contradictory condition in a given combination of FOSS and FRAND licence would prohibit a combination of the two works in a product. However, Blind and Boehm [15], as well as Maracke [94], and Phipps [115], point out that answering the question of legal compatibility is not a sufficient precondition for possible cooperation between SDOs and open source communities.

Blind and Boehm [15] find that the question of legal incompatibility can only be assessed in relation to a specific contractual situation and the individual conditions applied in the specific Open Source and FRAND licences. Legal compatibility checks only produce useful results in a specific licensing relationship with a specific Open Source licence in conjunction with specific FRAND terms. Even if a case does not show any incompatibilities, this only means that cooperation is legally possible, not that participants from SDOs and FOSS would be willing to participate and contribute to a common result. The legal compatibility of the licensing conditions is a necessary condition, but not sufficient to establish a successful cooperation between the SDOs and the FOSS communities.

IP regimes serve different purposes in SDOs and in FOSS communities. FOSS licences mirror and follow collaboration models. They represent how participants imagine the jointly created products to be used, resulting in a classification into strong-copyleft, weak-copyleft, and permissive FOSS licences. Governance within the Open Source communities initially developed as a model of cooperation and is then reinforced by choice of one or more licences.

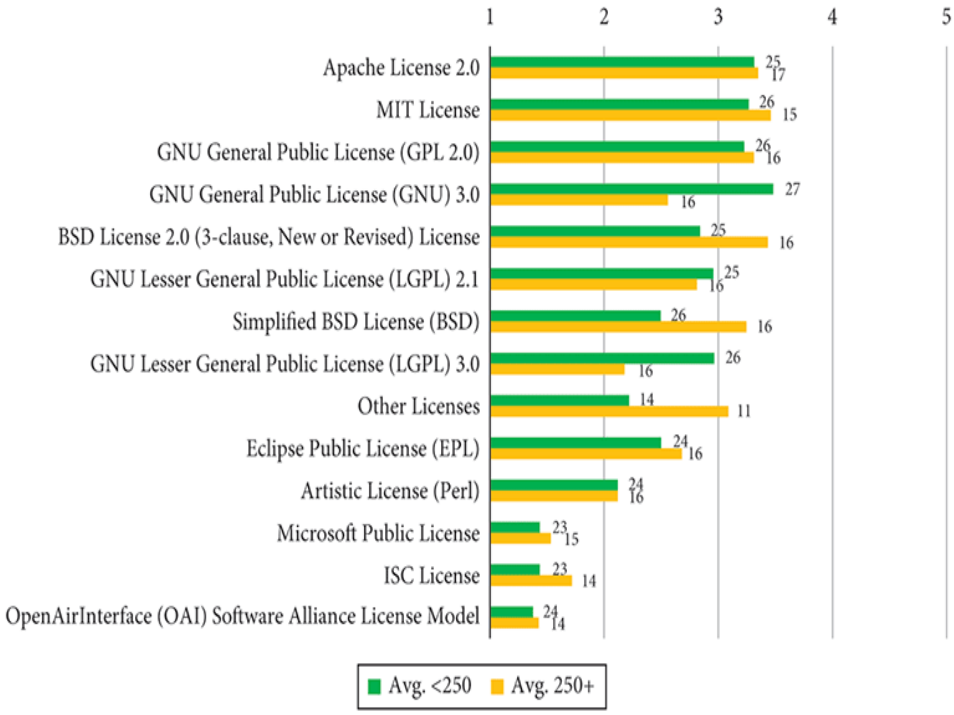
In contrast, the IP frameworks of SDOs regulate how participants engage and how conflicts are resolved. Special attention is paid to how participants can later withdraw from pre-competitive collaboration in SDOs and compete again in products that implement the developed standard. This rationale is alien to FOSS communities, as they do not intend to re-enter the competitive arena once a functional area is covered by an industry standard FOSS implementation. This contradiction could predict the idea that a FOSS community is creating reference implementations to a standard alongside other competing implementations. In the same context, FOSS communities see no benefit in participating in the development of standards only to facilitate alternative or competing implementations.

The investigation by Blind and Boehm [15] does not provide evidence that the limited cooperation between standards and open source development is caused by the uncertainty about the legal compatibility between SDOs and open source in the IP regimes. Most of the FOSS projects observed in this study use licences with reciprocal conditions or permissive licenses that include explicit patent grants and interacted productively with the SDOs relevant for their market segment. Some SDOs have responded to open source-related market changes by introducing flexible, or toll-free IP regimes such as W3C and OASIS, and adopting open source inspired methods of collaboration. The study by Blind and Boehm does not find a need to reconcile SDOs and Open Source IP policies.[15] With the exception of the telecommunications subsector, there seems to be no conflict between SDO IP policies and the policies applied to open source IP.

It seems that in practice participants adapt to the methods of cooperation and IP policies applied by the communities with which they cooperate, compromising between the contribution of their own IP and access to the overall contributions of the participants. For activities at the interface between standards and open source development, this usually means the introduction of a RF patent licensing policy for all examined cases except ETSI-NFV and OpenAirInterface, which actively anticipate the inclusion of FRAND-licensed SEPs in the developed standards. This expectation of RF licensing is considered acceptable and is not an obstacle to cooperation or the development of relevant standards.

In contrast to the distinction between freely licenced and FRAND used in standardisation, several licensing models have been developed for free and open source software (see Smith [135]). Looking at the most common regimes, the Apache Licene 2.0 (ASL-2.0), the MIT licence, and the GPL-2.0 are the three most commonly used in the study by Blind and Boehm [15], followed by the GPL-3.0 and the BSD 2-Clause "Simplified"

Figure 3.2: Participation in Open Source activities with various copyright licences—SMO vs LO (Scale: 1 = 'Never'; 2 = 'Rarely'; 3 = 'Sometimes'; 4 = 'Often'; 5 = 'Always') [15, p. 159]



License (BSD-2-clause). This ranking largely corresponds to the already publicly available data, which confirms both the validity of the selected cases and the representativeness of the sample. Less common are the GNU Lesser General Public License v2.1 (LGPL-2.1), the BSD-2-clause licence, the GNU Lesser General Public License v3.0 (LGPL-3.0), and the Eclipse Public License 2.0 (EPL-2.0).

In addition to the significant differences in the general attractiveness of the various FOSS licensing models, there are discrepancies between larger and smaller organisations. The latter prefer both the GPL-3.0 and the LGPL-3.0, while the former are inclined towards the permissive MIT and BSD licence families. In the case studies, however, it can be observed that many licence selections are made in the early phase of a FOSS project and never changed. This supports the assertion that licence choice in communities follows the collaborative model that contributors seek, and that newer projects with more corporate contributors more often choose licences with explicit patent clauses, as in the ASL-2.0 or GNU General Public License v3.0 (GPL-3.0) licences, as opposed to implicit or missing patent licence terms.

Overall, a framework for patent licensing has been established, either through the use of FOSS licences, which involve the granting of patents owned by the participants, or by requiring a declaration by the SEP or a commitment by the participants to patent licensing. Although host organisations allow an option for FRAND-based patent licensing in several cases, almost all cases have opted for a RF patent licensing policy. This is either because patents whose claims cover standardised functionality have expired, as in the case of C++, or because the working group aims to make the standard freely available, with the policy of no fees being implemented through a contributor licence agreement.

In case of conflict, the strict separation between FOSS and FRAND licences is still the preferred option, followed by negotiations to find solutions based on the experiences reported in specific cases. If no solutions are found, small organisations, in particular, will withdraw from standardisation. Another possibility is the use of pure copyright licences, which explicitly exclude patent licensing rights that are negotiated separately. Such licences are not recognised as open source licences according to the Open Source Definition. Sometimes, more flexible IP models are used in SDOs, which allow IP schemes and even withdrawal from FOSS on a case-by-case basis. This is less likely than withdrawal from standardisation.

While there are no convincing constructive solutions for conflicts between the licensing models in standardisation and open source development, some approaches to general cooperation between standardisation and FOSS are more promising, especially from the perspective of smaller organisations:

- First, the stakeholders call for greater flexibility in the patent policy of SDOs.
- Secondly, stakeholders ask for new processes for integrating open source development into standardisation.
- Thirdly, not only is a more flexible patent policy called for, but it is even proposed that SDOs change their patent policy in the direction of licence freedom.

- Fourthly, participation in the Open Invention Network (OIN) or similar patent cross-licensing agreements is used to resolve conflicts with respect to SEPs or FRAND licensing.
- Additionally, new governance and conflict resolution models evolve, for example the use of licences that explicitly exclude patent rights, or finally, a direct integration of SDOs and FOSS communities.

3.5 Conclusion

The development of the IP framework in the ICT sector is complicated by the ongoing hardware commodification, which leads to the predominant use of off-the-shelf general-purpose computers that integrate virtually all primary ICT functions such as computing, storage, networking, or telecommunications, and peripherals. Manufacturers face a market situation in which they need access to an extensive and comprehensive set of IP held by many different rights holders to produce competitive products. This means that standards and open source development must be analysed in combination when assessing governance norms and IP frameworks in the ICT sector. Market players are aware of this situation and have, in part, already adjusted by generally preferring consensus-oriented cooperation. Formal rules serve as a fallback for conflict resolution and may not form a practical day to day solution.

Compatibility between FOSS and FRAND licence conditions is recognised as a prerequisite for collaboration but is not communicated as a practical problem and is considered solvable. Legal compatibility is a necessity but not a sufficient condition for possible collaborations of SDOs and open source. Only a small number of licences are relevant in practice, which reduces the problem space for the analysis of the compatibility of FOSS and FRAND licences.

Regarding the existence of conflicts between the various copyright licences and the licensing models in standardisation, in particular FRAND, both the GPL-2.0 and GPL-3.0 and the LGPL-2.1 and LGPL-3.0 are mentioned by the majority of the stakeholders. Even with incompatibilities between FOSS licences and SDO IP frameworks, the stakeholders typically resolved these issues or worked around them driven by the common interest in the collaborative development of a standardised technology.

In case of conflicts, the strict separation between FOSS and FRAND licensing is still the preferred option, followed by negotiations to find solutions supported, and if no solutions are found, in particular small organisations withdraw from standardisation.

Another less popular option is the use of copyright-only licences explicitly excluding patent licence rights, which are negotiated separately.

SDO governance focuses on the legal and IP framework and is implemented within policy constraints in a self-regulatory manner, building on the basic policies defined in interaction with policy-makers (for an overview of different SDO IP governance models see the reference by Baron et al. [8]). FOSS governance is anchored in cooperation models and builds on as yet unregulated authority within the autonomous group of contributors. FOSS governance continues to converge, with volunteer-led communities relying on more implicit governance norms and industry-led open source communities creating more explicit rules in increasingly normalised project charters and governance structures. Governance in SDOs and FOSS communities still differ in key aspects of philosophy and implementation, which is a significant obstacle for collaboration.

SDO processes are inclusive in terms of involving a broadly defined group of stakeholders. They are also integrated into industry and policy-making. Open Source communities usually include companies, other organisations, and individual software developers without any systematic multi-stakeholder engagement. There is a strong overlap of participants in the development of standards and open source software, especially for large companies. Overall, FOSS communities apply a merit-based structure. They are less accessible to policy-makers and difficult to influence in line with industrial and innovation policy objectives.

Both SDO and Open Source communities are capable of small to large collaborations (in terms of the number of participants) and small to significant R&D investments. The open source umbrella organizations increasingly provide platforms for cooperation and consensus building, traditionally provided by SDOs. The broader use of implementation-first and parallel approaches to standardisation influences the utility of specifications concerning the value of common deployments. This changes the role of standards themselves, as standards and open source development become alternatives to achieve market dissemination for a technology. Open source collaboration is a new challenge for innovation management as it creates an innovative, state-of-the-art technology offered with the attributes of a consumer good, with the potential for accelerated mass adoption, which have traditionally been seen as opposites and public goods.

Globalisation and online collaboration are shaping the landscape of FOSS communities and SDOs to the extent that interactions are based primarily on relevance in the respective market segment and less on formal recognition. However, formal recognition still serves a purpose as it signals, for example, relevance for security standards and for a reliable basic policy accepted by policy-makers. The converging functions of SDO and open source umbrella organizations offer actors a choice of platforms that did not previously exist. Both approaches are successful in providing interoperability and competitive, innovative technical solutions. In both cases, access to a wide range of technologies needed to produce competitive products is key to the freedom of action of the implementers.

Three scenarios were observed: specification-first, implementation-first, and parallel standardisation. The specification-first approach is becoming less important in relative terms but is still essential in specification-driven technology areas. In particular, the parallel approach to standardisation represents some of the successful interactions between standards and open source development and can lead to higher quality standards, more innovation, and better implementation.

Innovation policy focuses on the framework conditions for IP and the orientation of research and development financing towards increasing competitiveness and promoting the development of technological champions and industrial competence areas. Open source processes represent a viable additional approach for the development of technical standards. The success of open source communities is driven by their dynamic capacity for innovation.

Chapter 4

Standard setting organizations and open source communities: Partners or competitors?

Standardization serves a as a means to improve our overall quality of life through the economies of scale gained from the pervasive adoption of technical solutions. It enables competition by facilitating interoperability between products of different vendors. The wider open source community develops FOSS in a global upstream/downstream model that similarly benefits society as a public good. FOSS and SSOs are both instruments causing standardizing effects. Innovators and policy makers assume that a mutually beneficial collaboration between them is possible. However, their exact relationship is not yet fully understood, especially when and how FOSS and SSOs complement each other, or displace each other as competitors. To be able to compare FOSS and SSOs, our study develops a phase model of standardization that is applicable to both approaches, and applies this model to compare the strengths and weaknesses of FOSS and SSOs against common opportunities and threats in the ICT sector. Based on qualitative expert interviews with FOSS and SSO representatives, the synthesis of the separate results support conclusions from a product, a process and a societal perspective. The study identifies *cost of change* as a key determinant for the efficacy of each approach. It concludes that FOSS and SSOs create complementary products, compete for efficiency of the standardization process, and are both independent and complementary standardization instruments available to industry and influenceable by policy makers. The paper closes with discussion of possible implications relevant to businesses, the wider open source community, SSOs and policy makers.

4.1 Technical standardization in the ICT sector

SSOs have a long-standing history in facilitating technical innovation dating back more than one hundred years [92]. The recognized positive economic, social and political impact of standardization resulted in strong ties between industry, government and research institutions and lead to the establishment of an integrated network of national standards bodies, European standards bodies, sector-specific SSOs and the ISO. Today, SSOs are an integral societal institution. We encounter standards in every moment of our daily lives. They increasingly define how we communicate with each other, and what quality to expect from our food, shelter, transportation, health care, banking, and environment. Standards developed by ISO, IEC and ITU in cooperation with national standards bodies define requirements for environmental protection[1], information security[2] or quality management[3]. Conforming with these formal standards means adopting an acknowledged state of the art or technical consensus. It is a common expectation by industry and consumers and potentially a requirement by regulators or public procurement.[154]

In contrast, FOSS is a relatively new phenomenon, even though it represents a tradition of collaboration in software development that predates proprietary software. It has gained widespread recognition because of its evident ability to produce high quality, interoperable and widely adopted software at a high pace of innovation.[83] The Linux operating system or the Eclipse family of software engineering products have become de-facto standards. FOSS success stories affect all walks of life: Firefox helps citizens protect their privacy. Android has been adopted by the mobile industry as the most widely used device platform. Much of the critical infrastructure[4] of the internet is based itself on the results of diverse par-

[1] `https://www.iso.org/protecting-our-planet.html` (accessed 03/02/2024)
[2] `https://www.iso.org/standard/27001` (accessed 03/02/2024)
[3] `https://www.iso.org/standards/popular/iso-9000-family` (accessed 03/02/2024)
[4] Open Source Security Foundation (`https://openssf.org/` (accessed 03/02/2024)),

ticipants voluntarily collaborating in communities on free software solutions.

Since both SSO and FOSS drive innovation [74], it is commonly assumed that "(i)ntegration between open source projects and standards development processes is a win-win situation: On one side the alignment of open source and standardization can speed-up the standards development process and the take-up of ICT standards (especially for small and medium-sized enterprisess (SMEs)) and on the other side standards can provide for interoperability of open source software implementations." [46] There are many concrete examples of successful collaboration between SSO and FOSS, especially in the development of internet and web technologies. However they rarely involve recognized formal SSO. There is also disagreement amongst practitioners whether or not FOSS achieves standardizing effects comparable to SSO, and is therefore an alternative approach to formal standardization. At the same time standards development and implementation are complementary and may benefit from collaboration between SSO and the wider open source community. Where and how SSO and FOSS can collaborate and where are they strategic alternatives with regards to facilitating effective standards is the central question this study attempts to answer.

We assume that by breaking down the alternative processes that create a standardizing effect in the case of SSOs and FOSS into more generic phases it is possible to create a model where SSO and FOSS activities that result in the diffusion of a standard can be systematically compared. By combining the individual author's perspectives of working in SSOs and of being a FOSS contributor as well as the perspectives of experts from SSOs and FOSS, we attempt to compare the strengths and weaknesses of the two approaches in a scientific, but also pragmatic and practically applicable way.

We approach the problem by developing a generalized standardization model that is subdivided into ideation, implementation, specification and diffusion as separate phases (see 4.3.4). We assume that both SSO and FOSS activities touch upon these phases, but not necessarily in the same order of events. Through semi-structured, qualitative expert interviews with representatives of SSO that are involved in evaluating the

previously referred to as the Core Infrastructure Initiative impact of open source approaches in their organizations and FOSS contributors with experience in standardization activities of their communities, we gather empirical data to perform two separate strengths, weaknesses, opportunities and threats (SWOT) analyses. The results are then contrasted in a comparative analysis to identify where the two constituencies complement each other, and where they compete for relevance. The results are discussed separately regarding the created products, the processes and norms applied in the organizations, and from a societal perspective.

We find that FOSS products commonly implement a multitude of formal standards and therefore complement the results of the work of SSOs. However we do also find that both represent systematically different approaches where participants choose between centralized and hierarchical as opposed to decentralized and meritocratic processes, and therefore represent competing mechanisms. Finally, we conclude that from a societal perspective, both SSO and FOSS approaches are effective standardization instruments, offering policy makers and enterprises a choice of which approach to support to drive innovation, to ensure competitive markets or to enforce regulatory goals.

4.2 Scope and literature review

4.2.1 Standardization and standards setting

Standardization is the process of creating specifications and implementations for technical solutions with the aim to reach wide adoption in the market. *Standard* in this paper means a technical standard that describes requirements towards technical systems, built of a combination of hard- and software.[17] Hardware refers to physical goods that require labor and materials to build, resulting in limited supply. Software are information goods that are "costly to produce but cheap to reproduce" [150], resulting in virtually unlimited supply. Related to computing technology, standards may describe the functionality of software systems (software standards), or the interaction between software or hardware systems (interface standards), or attributes of hardware (hardware standards). Technical standards that interact with the work of FOSS communities are either software standards or interface standards, not hardware standards. Communities may produce open hardware designs which are information goods,[5] or even engage in producing such hardware, however this is not the usual modus operandi and not discussed in this paper.

The ongoing commoditization of hardware fosters a shift from hardware standards towards software standards implemented using off-the-shelf components. The increasing relevance of software standards is evident for example in the transition to software-defined networking [43], or the use of container technologies in data centers that abstract away hardware interfaces all the way to current trends like server-less computing or function-as-a-service.

Ensuring interoperability, as in "the ability to transfer and render useful data and other information across systems [...], applications, or components" [57], is commonly identified as a central aim of standards setting. Specifications of data formats, hardware interfaces and communication protocols all enable interoperability between devices or programs following these standards. Both standards and FOSS development aim to achieve interoperability, using different and partially competing approaches. Systems can gain interoperability by either following a common *specification*, which requires a consensus based formal process and is the *formal standardization* approach of SSOs, or by using one or more common *implementations*, the *informal standardization* process which is closer to how FOSS communities approach the problem, or a combination of the two. By specifying first, standards enable implementers to create a multitude of standard-compliant products that are interoperable and compete in the market. Due to the relatively time-consuming formal standardization process, a formal standard is not changed or withdrawn quickly, which on the one hand offers investment security, but often lacks the flexibility to quickly react to technical advancements. By implementing first, FOSS products facilitate interoperability through a shared, freely licensed implementation defining a non-differentiating state of the art. SSOs facilitate a process where stakeholders agree on and formulate a written specification (the standard), and leave it to implementers to cre-

[5] See for example the LEON CPU architecture, which is licensed under GPL and commercial terms. http://www.esa.int/Our_Activities/Space_Engineering_Technology/Microelectronics/LEON2-FT (accessed 03/02/2024)

ate products that follow the standard after it has been set. FOSS communities apply a “code first” approach, where the expectation is that participants produce “working code”, and if necessary author a specification after a working reference implementation exists. Contributors usually do not create code with the goal of setting standards, however the various initiatives attempting to solve a specific technical problem typically converge towards a dominant design that finds wide adoption.[141] While SSOs often assume that a formal or de-jure standard is the precondition for standardization, FOSS communities achieve standardizing effects and create informal or de-facto standards by implementing first.

Because of this systematic difference in understanding, the standardization phase model used in this study to relate formal and de-facto standardization to each other does *not* assume that specification precedes implementation in the sense that implementation has to be based on an existing specification. Both formal and de-facto standardization are seen as comparatively effective paths to create a standardizing effect.

Literature and common practice differentiate between de-jure and de-facto standards, which differ in the way they are developed, and in the effects they have on the market. Formal (de-jure) standards are the result of committee work in recognized SSO. They are developed in a usually complex, consensus-driven process open to a variety of often diverse and competing stakeholders.[14] The process differs regarding the market and sector context as well as the often very different goals and strategies of the stakeholders involved.[52] The established platforms for multi-stakeholder elaboration implement a formalized and in part regulated process in which finding a common denominator may last several years. This stable, formalized process is a strength of SSO, as it alleviates anti-trust concerns and enables transparency, accessibility for all stakeholders and high quality standards. It is also a weakness, as SSO are in some cases not able to keep up with accelerating innovation cycles.[134]

More flexible informal standardization has evolved and broadened the standardization landscape.[156] By reaching a significant or dominating market share, technologies that are developed by individual market players, consortia or unrecognized SSO become de-facto standards.[117] De-facto standards emerge from competition in markets, industry alliances, market creating activities of intermediaries, technical specifications issued by influential market actors, by consumer choices and other factors.[7] Means like contracting, advertising and pricing help disseminate the technology at a large scale.[76, 138] De-facto standards are differentiated into private standards that are created by individual organizations and market standards that emerge through competition. Both are not initially defined by a formal SSO.[7] The fact that there is no formal definition of what a de-facto standard is and how it emerges adds to the complexity of a systematic assessment. De-jure standards are set by recognized formal national or international SSOs or through governmental regulation. Government agencies may ratify and SSO may adopt and formalize existing de-facto standards based on their acceptance in the market, which transforms them into formal standards.[7]

The unifying aspect of these approaches is that they primarily focus on the creation of specifications. The strict separation of formal and informal standardization has been questioned, underlining that actors chose between participation in SSO processes and in de-facto standards setting market processes and often developing a hybrid approach of parallel implementation and specification.[152] Participants have a choice between formal standardization, informal standardization and a flexible combination of the two. In this paper, the distinction between de-jure and de-facto standardization was found to be less important than the capability of a recognized or unrecognized SSO or industry consortium to produce specifications that are useful for the targeted market segments. There are far more consortia and fora than de-jure SSO. Some SSO support both accredited and unaccredited working groups. They are however renowned for the specifications they produce, and much less for their governance and formal recognition.

4.2.2 Establishing industry standards through joint implementation

The FOSS communities considered in this paper contribute to information goods, predominantly computer programs, in a collaborative process based on voluntary participation and often highly fluid and less structured than typical SSOs. Their products are public goods called free software because they are distributed under a license that provides the user with the freedoms to use, study, modify and redistribute these goods.[140] What these freedoms entail is detailed in the Open Source Definition as set by the Open Source Initiative. The commonly used term *open source software* originally describes a campaign to promote free software to business.[113] Both "free software" and "open source" (in this paper jointly and synonymously referred to as FOSS) today are proper terms of art that refer to software that is distributed under a license which complies with the Open Source Definition. The Open Source Initiative is the steward that approves licenses for being compliant with this definition.

FOSS *community* in this context describes a group of contributors and optionally their organizational entities that participate in the creation of specific free software goods voluntarily. Both individuals and entities such as businesses, universities and governments participate in FOSS production processes in a modus commonly referred to as peer production.[11] The social and behavioral norms and decision making processes applied by the group of participants are described by the governance of the community. Absent of an outside authority, governance is usually established by the group itself, with all authority born from within the community. Since all authority in the community is self-determined, behavior according to community governance norms set out informally or formally for example in *codes of conduct* is a prerequisite to participation. Violation of community norms is regarded as anti-social behavior.

While individual communities focus on the creation of specific software or solutions, there is a strong habit of inter-community collaboration, as evident in regular large-scale conferences like FOSDEM[6] that attract thousands of contributors from a variety of backgrounds. The common understanding between them is referred to as the *open source way* or sometimes as open source culture, resulting in an understanding of *the wider open source community* as an aggregate of the individuals and organizations participating in FOSS development with an interest to foster and protect it. The wider open source community jointly creates large and complex software products like Linux distributions that combine the products of possibly thousands of individual communities in the above sense. Prerequisite to this is a common understanding of how to create derivative and aggregate works that is based on the Open Source Defini-

[6] https://fosdem.org (accessed 03/02/2024)

tion and other fundamental open source norms referring to shared stewardship and access not only of the software source code, but also to the community development processes. Based on that the communities establish an upstream/downstream model of how their development processes interact with others, enabling cross-community collaboration in complex many-to-many relationships (see 4.5.2). Adherence to this common understanding is expected from all participating actors.

All FOSS licenses grant recipients of the software the rights to use, study, modify and redistribute it. Once a piece of software is released under a free software license, it will be free for everybody to use, forever.[51] While some free software licenses allow proprietary derivatives that themselves are not FOSS, the underlying work cannot be retracted from the market (or "un-open-sourced") once copies of it are in circulation. The usage rights embodied in FOSS licenses make the software non-excludable and non-rivalrous, so that the software becomes a public good through the application of the terms of the license.[154] Inherently, FOSS communities produce state of the art, but non-differentiating technology. The technology is state of the art if the community manages to develop new and innovative solutions. Competition for contributors and adoption drives communities towards excellence and leading technology to be adopted as de-facto standards. The technology is however not differentiating, because the software is available to everybody and therefore cannot differentiate between competing products. To differentiate, implementers need to combine the FOSS product in new and innovative ways or with other differentiating product features. All FOSS produced over time contributes to a common body of knowledge that grows steadily since none of it's elements can ever disappear. By competing for the adoption of their technologies, communities trigger fast innovation cycles. For that reason, the process that leads to FOSS is linked to innovation not just as a method, but also as a strategy.[129] Wherever the state of the art can be improved, existing solutions can expect to be challenged. Once a dominant design emerges, it is quickly adopted across the industry. It becomes a de-facto standard and a commodity. The FOSS community process is collaborative in organization, but competitive in nature. Since it produces an ever-growing commons of free technology goods, it is considered beneficial to the common good in various jurisdictions. Many community organizations like The Document Foundation or KDE e.V. are not just not-for-profit organizations but also recognized as charitable (in the United States trade associations (501(c)(6)) are more common).

Both SSOs and FOSS communities differentiate between their products, the outcome of their activities, and their processes, which describe how these outcomes are created and agreed upon. The European Interoperability Framework for example imposes requirements for open standards regarding the standards setting process ("all stakeholders have the opportunity to contribute to the development of the specification and a public review is part of the decision-making process"), and the licensing of the product ("intellectual property rights to the specification are licensed on FRAND terms, in a way that allows implementation in both proprietary and open source software, and preferably on a royalty-free basis").[47] In this study, the interaction between SSOs and FOSS communities will in a similar fashion be reviewed separately from a product and process perspective.

4.2.3 Existing interactions between Open Source and standardization

Research on the interaction between FOSS and SSO focuses on the compatibility of free software licensing terms with those of standards. A key aspect in this debate is the question of under what conditions a technical standard may be implemented as FOSS. Lundell, Gamalielsson, and Katz [91] conclude that uncertainty about the identity of rights holders and the patent claims reading on the standard are associated with significant risks for implementers of FOSS, and recommend a royalty-free licensing option of SEP for free software implementations. The discussion about what constitutes an *open standard* focuses on the relationship between the licensing terms of the specifications of the standard and related SEP and the viability of the FOSS model of collaboration under these terms. While it seems to be accepted that "(t)he availability of an OSS implementation will spur quicker adoption and acceptance of the standard as everyone has easy access to the implementation of the standard and so can try and test it out" [3], there is no agreement yet on the licensing terms required to facilitate free software implementations. It can be construed that specification and implementation should be considered separate in terms of licensing policy, however practices like automatic sub-licensing of derivative works and the absence of license negotiations and monitoring of individual licenses or licensees are considered essential to the FOSS model of collaboration.[59]

Cooperation between SSO and FOSS appears desirable because of the fast pace of innovation in free software and the obvious utility of an ever-growing body of public software goods. Compatibility of licensing terms is only one aspect of enabling this cooperation. In contributing to FOSS projects, participants do not contribute with the expectation of obtaining future license revenue from that contribution. Actors contribute voluntarily following their self-interest and on the basis that others will be contributing on a similar basis. Information asymmetries regarding access to the underlying technology as well as differences in cultural norms and expectations may affect their motivation to do so, possibly to the point of inhibiting successful collaboration. This may require facilitation, incubation of projects and coordinated policy making to achieve successful cooperation between SSO and FOSS.[107]

Overall, while the question of legal compatibility has been discussed extensively, and the usefulness of collaboration between SSO and FOSS communities has been established, there is not much research at this time on what benefits both sides would realize from such a cooperation. Successful examples exist that represent a special case of standards competition.[13] The process of developing the Open Document Format for Office Applications (OpenDocument) standard brings together community actors and industry at OASIS, the results are later formalized at ISO.[102] The IETF considers itself an *open standards organization* without formal membership, and develops internet and network standards like TCP/IP or the widely used Secure Shell (SSH) protocol. W3C specifies formats like CSS and XML with widely deployed FOSS implementations like Webkit or Gecko. Open Source Mano is an ETSI-hosted initiative to develop an Open Source NFV Management and Orchestration (MANO) software stack.[43] Collaborations of this kind are however not ubiquitous or common. This paper will investigate what promotes and what inhibits larger-scale cooperations and discuss potential implications for SSO, FOSS, regulators and enterprises.

4.3 Model and objectives

A fundamental research objective of this study is to establish a generalized phase model of standardization that can be applied to various standardization instruments, including formal standardization in SSO as well as FOSS processes. De-facto standardization follows a relatively undefined process, while formal standardization processes are usually well-defined and strictly followed. By applying a common phase model of standardization, SSO and FOSS activities that cause a standardizing effect become comparable. The phase model supports identifying activities where SSO and FOSS complement each other and where they are in competition. This model is then used to describe differing approaches of standardization efforts that are identified during qualitative interviews. Finally, we aim to discuss possible implications relevant to policy makers, businesses, SSO and FOSS communities on how position and adapt their organizations.

4.3.1 Early and late standardization

Most definitions of standards, including that applied by ISO, emphasize both the document character and the formal approval processes for it to become effective. ISO/IEC Guide 2:2004 defines a standard as "a document, established by consensus and approved by a recognized body, that provides, for common and repeated use, rules, guidelines or characteristics for activities or their results, aimed at the achievement of the optimum degree of order in a given context." By including features that conform to such standards, products from different companies can benefit from network effects. A common thought model is that the standard is embodied in a specification, and that standard-compliant products implement this specification. For de-facto standards, these events do not necessarily occur in that order. Products may be implemented, introduced in the market and become successful with a specification added as an afterthought. This indicates that innovations that become standards will exhibit *implementation* and *specification* phases, however the order is not predetermined. We use the term *early standardization* if the specification is created first and then implemented, *late standardization* if the implementation is created first and then formally specified and *parallel* *standardization* if specification and implementations are created at the same time.[90]

This observation leads to the hypothesis that processes that cause a standardizing effect proceed through a set of relatively generic phases, and that different approaches to standardization differ in the order of events in which those phases occur. A *standardizing effect* on a market segment, an industry sector or society as a whole in this context describes the transition caused by a standardization instrument from a situation where a need for standardization is perceived to a situation where that need was satisfied by the widespread diffusion of a standard. The need for standardization may arise from various governmental, societal or market motives.[17]

These instruments cause a standardizing effect by influencing market actors to adopt common technical solutions. The influence could be manifested in an intervention through authority, for example in the form of governmentally enforced safety standards, or in an extreme case through the order of a dictator. Compared to that, formal standard setting in SSO is preferred, as it replaces the need for an intervention through authority with a consensus- or market-driven platform based on voluntary participation and involving a variety of stakeholders.

Other instruments exist next to formal standardization in recognized SSOs. Standardizing effects are also caused by normalized customs and practices enforced by tradition, self-regulation, codes of behavior that are prevalent in some industry sectors, especially trade, but also industrial consortia, professional charters, or FOSS governance.[72]

To avoid the nomenclatorial confusion that stems from the customary use of the term "standard" in the sense of an approved document as well as a widely adopted technical solution, we will use the term *standard* for a dominantly used solution to a technical problem, the term *specification* for the document that describes this solution, and the term *implementation* for a product that embodies the standard according to the specification. Other interpretations of these terms are possible. For some readers, the term "standardization" typically refers to the technical specification development process. However, standardization may be achieved in different ways, sometimes completely without a-priori specification. It would defeat the purpose of this study to reduce the understanding of the term *standard* to a formalized written specification. It would assume that specification precedes implementation, and exclude most de-facto standards, especially FOSS products.

4.3.2 Interoperability in software and interface standards

Especially de-facto standardization does not follow a common model that may be easily recognized or replicated by competitors. Market forces determine the success of a de-facto standard based on the availability of products and technology, marketing skills and timing. Any attempt to put together a recipe with these ingredients would end in speculation.[137] In relation to FOSS, this difficulty in describing a common model of the development of de-facto standards is aggravated by an observable shift in what is being developed as a standard from common data formats to communication protocols to joint implementations.

In reaction to the dominance of specific applications in the ICT sector, the wider open source community initially approached the problem of recreating a competitive environment by developing *standard data formats*, and lobbying for dominant commercial products to support these. The OpenDocument standard opened the market for office productivity suites by challenging Microsoft Office.[102] The existence of such strong de-facto standards implemented in highly concentrated dominant proprietary applications may be attributed to the explosive historical development of the personal computer market.

Widespread use of networked applications increased competition in the ICT sector by reducing the dependency on specific file formats. Instead, interoperability was achieved by enabling the communication between different applications through *standardized protocols*. A relatively large and diverse number of stakeholders was involved in creating these protocols as open and royalty free standards, for example during the developing of the XML standard at the W3C.[153]

With the wider adoption of open source software in enterprise contexts, more advanced forms of industry collaboration established themselves. Open source focused umbrella organizations like the Linux Foundation or the Eclipse Foundation enabled direct, continuous collaboration on the development of FOSS in large consortia of otherwise competing participants. The resulting body of non-differentiating commodity software today provides the foundation for proprietary commercial products built on top of them. The focus on standards setting shifted again from protocols to *joint implementation*. Today, many of these consortia embrace a "code first" philosophy and consider specification not only as an afterthought, but as a reason for *community fragmentation*. Fragmentation refers to a situation where the wider community develops competing implementations of the same technical solution, leading to a race for adoption and fewer contributors to each individual single product. This is considered a waste of resources, since in FOSS one jointly developed implementation is available to everybody, and competition for market share as such is pointless in the commons.[111]

4.3.3 Continuous non-differentiating cooperation

Instead of competing for market share with standards-compliant products by different vendors, FOSS solutions compete for adoption in the upstream/downstream model and for contributors. It is assumed today that computing devices ship with the larger part of their software stack based on collaboratively developed FOSS solutions. This approach to developing products that combine FOSS and proprietary source code, sometimes referred to as the "Pareto rule of software", enables product developers to invest the majority of their research and development spending into differentiating product features.[94]

The resulting model of collaboration on joint implementations between otherwise competing market actors systematically differs from earlier approaches, especially pre-competitive R&D cooperation. Previously, industrial consortia established elaborate terms of reference to be able to engage in pre-competitive cooperation without raising concerns about possible collusion in the eyes of anti-trust authorities. Under the *continuous non-differentiating cooperation* model, enterprises now commit to continuous collaboration on the non-differentiating elements of the software stack with their competitors. Continuous non-differentiating cooperation is a collaboration model implemented in FOSS communities and facilitated by FOSS foundations. It enables otherwise competing market actors to continuously cooperate to develop a common software stack that serves as basic, non-differentiating technology prerequisite to products that combine free and proprietary software. In contrast to pre-competitive cooperation on the development of proprietary, differentiating products, anti-trust concerns are not relevant in the continous-non-differentiating cooperation model since collusion is impossible if the results are immediately available to the general public and the development process is generally open for participation to all interested parties. The possibility of forks limits the control individual entities can exercise over the development process..

Continuous non-differentiating cooperation significantly reduces the transaction cost of collaboration due to non-negotiable free software licensing terms, and eliminates welfare losses caused by the parallel development of products in a standards race where later only one will become an adopted standard. Implementers perceive a strong motivation to participate in the collaborative effort, since being able to share the majority of the software stack becomes a prerequisite to producing at similar cost as competitors and to competitive product pricing.

The emergence of this new model of collaboration influences the efficacy of standardization instruments. The utility of specifications is reduced if adopted standards are based on a single, quickly evolving joint implementation. Where such a development model is applicable, as for example with the Linux kernel, interoperability is a given from the start, so that specification does not serve that purpose anymore.[35] "As a 'code first' organization, AGL members are collaborating to build a brand new Linux-based software platform and application framework that serves as the de facto standard for the automotive industry. Adopting an open platform across the industry enables automakers and suppliers to share and reuse the same code base, which will reduce development costs, decrease time-to-market for new products and enable rapid innovation across the industry."[7] In such scenarios, the product source code serves as the documentation. Specification may still be useful, however it needs to provide utility beyond interoperability, for example to document safety norms or other requirements.

[7] https://www.automotivelinux.org/software (accessed 03/02/2024)

4.3.4 A phase model of standardization

While the process of formal standards development varies between different SSOs, it commonly proceeds through a combination of proposal, preparatory, committee, inquiry, approval and publication stages.[145] Sometimes one or more proposals are submitted for a ballot vote. After identifying the proposal with the most support by experts, a working group discusses the suggested approaches to create the new standard. A draft is published for the public to comment on. The comments are discussed in the working group and a final version is being published.[145] From this point on the standard is promoted and diffused into the market in order to gain wide market acceptance.

Structured processes of developing new technical solutions and become de-facto standards regularly exhibit the phases of strategy development, idea generation, screening and evaluation, business analysis, product development, market testing and commercialization. These phases were adopted and modified throughout literature over time, and eventually condensed into three main steps, pre-development activities, product development, and testing and commercialization.[151]

Based on these two descriptive models of standardization and product development processes, a technical specification that also is implemented and adopted in the market will iterate through four basic phases - ideation, specification, implementation and diffusion. The order of these phases depends on the strategies chosen by the stakeholders involved (see 4.3.1).

The starting point is marked by a perceived need for standardization that triggers a strategic decision of stakeholders to participate. The need may be caused by a lack of acceptable solutions at the micro level of individual actors, inefficiencies causing friction between actors that require interoperability at the macro level of markets or industry sectors, or by societal needs like requirements for safety or security standards. Micro and macro level needs represent a market pull for standardization, while societal needs often indicate a regulatory push. In any case, a need for standardization causes market participants to initiate the development of a technical solution by entering the *ideation* phase. During the ideation phase potential specification or implementation approaches and solutions are put forward, analyzed and evaluated until a promising initial concept has been identified. The process of de-jure or de-facto standardization begins with a proposed technical solution to a perceived need for standardization. This understanding is in line with the classic concept of the relationship of requirements, needs and demand in economics.

At this point the paths to an adopted standard already diverge. If the actors involved choose a formal standardization strategy, they enter the *specification* phase. Facilitated by a SSO, and applying the structured, disciplined and transparent process defined in their terms of reference, a specification for the wanted technical solution is authored and formally approved. Once this specification, the formal standard document, is published, it enables manufacturers to enter the implementation phase and compete with standard-compliant products that satisfy the original need for standardization. ITU-T V.24 is an example for a formal technical standard that facilitates price competition by a multitude of competing implementers.[71] This approach describes the *early standardization model*.

Alternatively, vendors may enter the *implementation* phase immediately by developing products and introducing them into the market without prior specification. The resulting products will not be systematically interoperable, even though market pressure may coerce vendors to afford their consumers a certain level of integration with competing products. Consumer demand and the assumed original need for standardization may motivate vendors to create formal specifications after their products have been introduced. By implementing first and specifying later, these vendors have chosen a *late standardization model*, as is common for FOSS communities. The *Open Container Initiative* for example was created after the dominant market position of the Docker container engine manifested an industry need for re-enabling competition through specifications of container runtimes and related products. It describes itself as "an open governance structure for the express purpose of creating open industry standards around container formats and runtime".[105]

The process of technical standardization ends in adoption of one or multiple solutions to the original need for standardization by the demand side of the market. Once standards compliant products have reached *diffusion*, market participants engage in a combination of price competition for the standardized product attributes, and feature competition for their differentiating attributes. The success of their products will manifest itself in a weak or strong market position. In the ICT market, especially for network and platform products, network effects may lead to the market "tipping" towards one dominant supplier, and to their product emerging into a de-facto standard.[133] Independently of which strategy actors choose to convince the market to adopt a specific solution, the standardization process for this specific need ends when one solution has become the adopted standard.

In reality, the process of developing a specific specification and the corresponding implementations may not be so clear-cut.[34] SSO differ in the rigidity of their processes and their formal recognition. Businesses may opt to engage in the implementation and specification phases at the same time, for example to be able to create products while the formal standard is being developed or to build up IPR portfolios while advocating for covered functionality to become part of a formal standard. Some organizations strategically decide to participate in standards development to decrease market uncertainty or to ensure conformance with later government policies.[17] As a third alternative next to early and late standardization, these actors have chosen a *parallel standardization model.*[90]

The four phase model developed in this study is able to describe a variety of early, late or parallel real-life standardization processes. Since FOSS initiatives commonly follow a late or parallel standardization model, the model will serve as a reference for a comparative analysis of SSO and FOSS based standards development. The key question to answer based on this model with regards to the transition from ideation to market diffusion is what determines the choice for early, parallel or late standardization.

4.4 Qualitative methodology based on experts interviews

Assuming as derived in the standardization phase model that SSO and FOSS are both alternative and viable strategic paths towards an adopted standard, a method was devised to gather data about the researched subject matter that reflects the still amorphous structures in the field of study. Both sides of the story represent a grown, established culture difficult to describe in quantitative metrics. Research for this paper was conducted in four steps. First, data was gathered through a series of qualitative expert interviews. Second, a model of common opportunities and threats caused by major developments in the ICT sector that affect both SSO and FOSS was created. Third, two separate SWOT analyses where conducted to structure the findings for both fields individually. Finally, a comparative analysis was performed contrasting the two SWOT results to identify possible areas of competing and complementary activities between SSO and FOSS.

4.4.1 Interviews process and interviewee selection

The data for this study was gathered by performing a series of qualitative, open ended, semi-structured in-depth expert interviews according to the Meuser-Nagel method.[147] The chosen interview method focuses on making specialized expert knowledge accessible outside of their specific circles. Based on its recommendations, the interviews have been prestructured to map out and structure the field of interest, with the individual questions being defined openly to leave room for a open and flexible course of discussion. This allows the interviewee to present a personal view of the subject, and avoids the potential collapse of communication possibly caused by an over-structured interview.[96]

One possible risk with performing semi-structured qualitative interviews is that the interviewer must be capable to express his interest for a specific knowledge of the expert and at the same time be a competent dialogue partner at eye level. To mitigate that, the interviews have been conducted by two researchers with a distinct background and professional experience in both SSO and FOSS.

The interviewees have been invited to participate in the study based on their specific experience at the intersection between standards setting and FOSS development. We interviewed SSO representatives who are responsible for establishing open source policies for their organizations, especially ETSI, Deutsches Institut für Normung e.V (DIN) and Institute of Electrical and Electronics Engineers Standards Association (IEEE-SA) as well as FOSS community representatives in standards development at OASIS. The interviewees where all long-term engaged in the subject of the study and provided well-reflected insight based on their experienced. The level of experience provided by the interviewees was crucial for the overall success of the study. Each interview lasted between one and two hours.

The interviewees agreed that the interview will be recorded, and the audio archived at the chair of innovation economics of the Technical University of Berlin (see 4.9). They agreed that the interviewees will be named in the paper, that all statements will be aggregated or anonymized, that no individual statements will be published and that any potential citations will be approved before publication. The interviews were conducted live in an office surrounding, via internet based video conference or as a internet based telephone conference. The circle of participants always consisted of the two researchers and the interviewee, in order to ensure that both researchers with their FOSS and SSO backgrounds could interject questions to make sure all views and aspects are captured. The interviews were not interrupted which allowed the interview to develop and create a unique dynamic, without losing direction. The interviews were digitally recorded and afterwards transcribed.

4.4.2 SWOT analysis applied to SSO and FOSS

A SWOT analysis is a strategic planning tool that is based on the understanding that the successful performance of an organization with respect to a specific goal depends on the way the organization interacts with the inherent characteristics (internal factors) of the organization as well as the broader context (external factors) in which the organization must act but has no direct control upon.[68] It is employed to evaluate the strengths, weaknesses, opportunities and threats of an organization with regards to a project, a strategic approach or any other situation that requires a managerial decision. Strengths are defined as capabilities that enable an organization to perform well. These are capabilities that should be leveraged. Weaknesses are defined as characteristics that prohibit an organization from performing well and need to be addressed. Opportunities are defined as outside trends, forces, events and ideas from the organization external environment that the organization could capitalize on. Threats, in contrast, are possible events, forces and trends outside of an organization's control that the organization needs to plan for or decide how to mitigate.[42]

The approach is to analyze the external factors of opportunities and threats as well as the internal factors of strengths and weaknesses and to structure them into a matrix, which will then allow the derivation of strategic consequences. This matrix is commonly called “the SWOTs” of an organization with respect to a specific topic. It serves as input to strategy development by identifying whether there are strengths that can help capitalize on opportunities, strengths that can offset weaknesses, or opportunities that will offset threats. This process can be repeated for the different SWOT elements that have been isolated in the process. This list of possible strategic measures is then clustered and prioritized in order to identify feasible and effective strategic measures.

There is no common model to describe how two groups of organizations can successfully collaborate. To answer this question for SSO and FOSS, we devise a method to combine and contrast two separate SWOT matrices into a comparative analysis that highlights competing and complementary strengths and weaknesses of the two approaches. We believe that this approach is new and corresponds directly to the research question of finding areas of collaboration and competition between SSO and FOSS. The approach assumes that both organizations are operating in a *common* space of technical innovation with the same underlying trends, which means that two of the four dimensions of the separate SWOT analyses, strengths and weaknesses, are independent, while the other two, opportunities and threats, describe the same space. A strong overlap between the opportunities and threats identified is to be expected in the results, even though they may be perceived to have dissimilar effects. The comparative analysis is performed by evaluating, against every identified opportunity or threat, whether there are strengths and weaknesses of both camps that *negate* each other, indicating an area of potential competition, or that *reinforce* each other, indicating an area of potential collaboration.

An analogy can be drawn between the idea of comparative SWOT analysis and the economic theory of comparative advantage, where two countries mutually benefit from each other through specialization. It is also possible to extend the approach to a “multilateral” SWOT analysis where the cross connections between the organizations increase in complexity and generate exponentially more potential strategic ideas. However the conceptual extension of the original SWOT analysis serves specifically the purpose of contrasting two disjunct constituencies and depends on them operating in the same space of technical innovation.

4.5 Empirical analysis of the interview results

4.5.1 Threats and opportunities in the ICT ecosystem

With regard to the subject of this study, SSO and the wider open source community operate in the same space of technical innovation. Both are influenced by threats and opportunities caused by market trends, technological developments and long-term societal paradigm shifts that are, at least in the short-to-medium term, outside of their own control. The widespread use of computers and the creation of the internet represent *digitalization* and the emergence of the *digital society*. While other recent technical progress appears evolutionary, digitalization has the impact of a radical innovation at global scope. Digitalization is a revolution.[158]

The essential effect of digitalization in our context is the drastic reduction of transaction costs incurred from collaboration and information sharing. Three significant paradigm shifts related to digitalization have been identified in this study: the development of better methods of collaboration, a general trend towards openness and transparency, and a shift of relevance from national to supra- and international collaboration and regulation. A fourth influential trend is the shift in understanding of the role of the modern state. "Formerly conceived in many countries as a provider of jobs through the civil service and a producer of goods and services through public enterprises, in its modern form the state ideally sets the rules and intervenes to correct market failures, rather than substituting itself for the market as a mediocre manager of enterprises."[144] While they may perceive the effects of these paradigm shifts differently, both SSO and the wider FOSS community respond to them and in the interviews considered them as key factors in their interaction with each other and with society.

The *development of improved methods of collaboration* is directly related to the use of the internet as a means of direct user-to-user communication. Online collaboration tools remove the need to publish and distribute physical documents, facilitate participation independent of physical vicinity, and enable levels of transparency that where previously impossible due to the prohibitive cost of participation and sharing information. Detailed formal standards are useful in a workflow where these standards authored up-front in direct collaboration become the specification for the following independent development of conforming products. This workflow was effective because direct collaboration was costly and time-consuming. Its benefit is less clear today, especially in the realm of software and software related technologies, where permanent online collaboration and the pervasive availability of all relevant documentation are the norm. Less centralized and more inclusive decision making models continue to emerge, making formally and informally regulated committee work and the level of trust and authority embedded into the appointed committee members increasingly redundant. This change may be perceived as both an opportunity or a threat depending on the extent to which working in formally appointed committees is central to the culture of the organization, and possibly embraced by regulators. Some SSO like IEEE-SA reacted by decentralizing their committee work. FOSS communities are generally organized in a more or less decentralized way. Other stakeholders struggle with the perceived loss of control. Regulatory influence is especially challenged by the loss of relevance of national legislation. Meritocracy in online communities means that an appointed representative of a large industrial country should not expect to have more influence than a representative of a developing country, or a senior engineer more than a contributing young developer. Opportunities for regulatory capture and rent-seeking become increasingly rare, while stakeholder diversity increases. These opportunities towards a more balanced global industrial innovation process will be seen as a threat by those entities that benefitted from the old ways. SSO depend on new technical solutions and qualified experts engaging in their formal standardization activities. Developing market relevant standards requires innovative committee members as well as access to cutting edge technology championed by participating member companies. This challenge for SSO requires them to re-think collaboration and committee work patterns and opens an opportunity to collaborate with innovators in the wider open source community.

New possibilities of information sharing drive a major *trend towards openness and transparency*. What started with FOSS as a movement towards essential user freedoms [140] spread to *open access* in science, *open data*, *open organizations* [155], more inclusive political platforms and numerous other areas. Existing institutions have partially developed under the assumption, as in representative democracies, that direct, decentralized and pervasive collaboration between all stakeholders is practically impossible. However through digitalization the technical means for such decentralized collaboration now exist, resulting in an emerging expectation that transparent and open collaboration methods should be applied more widely. FOSS communities are one example of new institutions that developed to match the paradigm of transparent collaboration.

Deeply embedded into the culture of SSO are rules for the handling of IPR both towards the standards as products in their own right as well as to inventions embodied in the specifications of the standard. It is common, for example, that standards defined by DIN are sold by the copy through a publisher. Detailed IPR licensing policies of SSO regulate how SEP are made available to implementers. Although many SSO consider themselves to be transparent, inclusive and open, the economic models of SSO are closely intertwined with notions of exclusivity, linking the trend towards openness to a perceived conflict between exclusivity and open collaboration. Exclusivity can result in an advantage for players with earlier access to information than their competitors. SSOs can offer their members market influence, control over IPR and time-to-market advantage, which is why formal standardization resonates with industry as well as regulators. Even though the disconnect between exclusivity and the trend towards openness has been described as a conflict between incumbent businesses and FOSS participation during some of the interviews, many businesses also state that they are successfully combining business and FOSS activities. The number of enterprises engaging in collaborative projects at FOSS umbrella organizations like the Linux Foundation indicates there is no general difficulty for enterprises to adopt open and transparent collaboration models, and that the reported difficulties are most relevant in specific industry verticals where the traditional model of standardization in combination with IPR licensing is most beneficial to the participating companies, in particular the telecommunication and mobile communication sectors.

The debate at the EC level about the role of patents, SEP and FRAND licensing in ICT standardization demonstrates a strong stakeholder polarization, preventing the adoption of compromises even though both opportunities and threats present themselves to all involved parties. This intense influencing of policy makers and established institutions like the European Patent Office (EPO) is perceived by the wider open source community as a threat, namely the repeating pattern of using other IPR or contractual agreements to diminish the user rights granted by FOSS licenses. In the past, contributor license agreements (CLAs), trademark licensing programs, trade secrets (in the context of open source, for example, in the form of delayed source code releases), SEP in combination with FRAND licensing programs and other means have been applied to maintain a privileged position of one entity at the cost of software freedom. Other actors have attempted to redefine what the term open source stands for with the goal to prevent reuse of the source code or the development of derivative works. Activities like this are known in the wider community as spreading "FUD" (fear, uncertainty and doubt) and are considered a threat to the viability of the upstream/downstream model. Usually, these attempts are not successful, and software related businesses attempting them expose themselves to ridicule.[8] To be able to publish relevant, adequate and timely standards, SSO regard their IPR licensing and SEP policies as an opportunity that incentivizes innovators to secure a return of investment on research and development. Other technology developers may hope to establish a de-facto standard by offering their technical solutions under a FOSS license, reducing their motivation to participate in standards setting at SSOs. The convergence of technologies, digitalization and the trend towards openness increase the importance of interoperability and interface standards. Participants have a choice to collaborate on creating specifications, joint implementations or both. This presents an opportunity as there is an increasing demand for standards development that SSO can respond to. SSO have the opportunity to facilitate standards development and to provide the collaboration platform for the interaction between standards and FOSS development. FOSS communities are particularly sensitive to the perceived dangers of SEP

[8] https://opensource.org/peru_and_ms-php (accessed 03/02/2024)

claims covering their products, and the potential moral hazards originating from information asymmetries between patent rights holders and the implementer community. This perceived threat is aggravated by the lack of definition of what FRAND stands for, and the unpredictable behavior of non-participating entities that are not bound by SSO licensing policies asserting rights against joint implementations. The wider open source community is however usually confident in their own innovation capabilities which reduces the threat of non-participating entities gaining valid patent grants that cover essential FOSS technologies.[15]

The *shift of the role of the modern state from an employer and producer to a regulator* creates a new balance between societal or regulatory interests and industry.[144] Formal standard setting gains importance in its role as a provider of frames of references and rules that regulators can use to set directives or framework contracts. In this context, standards support certainty in compliance and legal matters by documenting expected behavior. This is an opportunity for national standards bodies that are integrated into regional or international cooperations of standards bodies since their remit matches the regulatory reach. This trend reinforces expectations towards the openness of standards. The European Interoperability Framework requires for open standards to give all stakeholders the opportunity to contribute to the development of the specification, the availability of the specification to everybody to study, and for the relevant IPR to be licensed on FRAND terms in a way that allows implementation in both proprietary and open source software, and preferably on a royalty-free basis.[47] There seems to be a general expectation that if regulators mandate compliance with a standard, then that should be an open standard according to these principles. This attitude is related to the general trend towards openness and collaboration described above. Similarly to standards, an expectation develops that software developed using public funds should always be released under a free software license. The public money - public code (PMPC) campaign is a community effort lobbying for this approach.[55] The stronger focus on regulation bears opportunities both for SSO and for FOSS communities. One potential mismatch exists between technical evolution and social responsibility that regulators need to take into account. It is common that SSO enter into treaties with the state or are entirely state run, agreeing to obligations to include a wide range of stakeholders into standards setting processes as well as to keep social responsibilities into account. This existing institutional integration between SSO and the public puts SSO in a unique position to develop standards that then become mandated. The self-regulated FOSS community is not only not obliged to keep social responsibilities into account, it often explicitly refuses to be influenced by stakeholders that are not actively participating in the development of their products. Voluntary participation is an effective regulator to ensure that the communities themselves are open to a diverse range of contributors. It does not however steer communities to keep outside interests in mind. Since the wider open source community causes an important and noticeable impact on society, an articulated public interest in the development of free software exists that regulators may eventually act upon. This can be understood as the risk of a potential disruption of established norms of collaboration of the wider open source community that until now primarily self-regulates. Most communities are not fully aware or acceptant of this possibility.

The final major change identified in the interviews is the *shift of relevance from national to supra- and international collaboration and regulation* caused by globalization. It is obvious that the benefits from standards increase the wider they are adopted. The various shapes of power plugs are a regular reminder that SSO originally developed in their countries, and only later started collaborating. It is also obvious that the parallel development of technical standards in different national standards bodies constitutes a welfare loss similar to parallel invention in patent races. Interaction with regulators in Europe however happens at the national and the EU level. The resulting need for new boundaries of responsibilities is both an opportunity as well as a threat for SSO. The development of standards may shift towards international, possibly sector specific standards development organizations, while the dissemination, translation and adoption of standards to a country's environment may remain the responsibility of national standards bodies. National borders never played an important role for FOSS communities, even though cultural barriers do. Cohesion within the wider community is strong and clusters in regions with a common cultural and language background. There is, for example, little interaction between the Chinese and the European FOSS community. Global collaboration is understood as an opportunity for

FOSS communities, even though their international integration is not yet perfect. Businesses participating in standards development and in FOSS activities at a significant level usually operate in multiple countries and benefit from the reduced barriers to market entry caused by the convergence of technical standards. Regulators at the national level may find it difficult to exercise influence over de-facto standards development, and may even find themselves facing a competition for defining standards that reduce their ability to implement their responsibilities towards their own constituencies. This is already apparent in the difficulties in regulating large internet platform businesses based in the US or China. ISO, the SSO recognized by the EC and the national standards bodies need to find a new balance of shared responsibilities that matches their social responsibility at each level.

4.5.2 Strengths and weaknesses of FOSS

The strengths of FOSS emphasized as most relevant with regard to this study during the interviews are the application of rapid prototyping, a global development model based on early and regular releases, voluntary participation in community activities, and the established overall development process of the wider open source community community in a complex upstream/downstream network facilitated by community umbrella organizations, especially FOSS foundations. Weaknesses are a lack of established supply chain management processes, uncertainty and arbitrariness in license compatibility and achieving license compliance, and a meritocracy that focuses primarily on product contributions at the neglect of other potential stakeholders.

Rapid prototyping refers to the ability to quickly sketch solutions to upcoming technical problems, and evaluate them based on the response of other participants. This enables a competitive evolutionary selection of solutions in an early stage, limiting costly up-front investments. It also facilitates challenging existing de-facto standards, reducing the hysteresis caused by an existing, outdated solution. As a result, FOSS market segments are “tippy”, they tend to quickly replace even widely adopted solutions that many actors are invested in if a new, more promising solution appears more convincing. A recent example is the decline in relevance of the OpenStack project that has seen massive investments by a large number of actors in 2016 once more efficient container-based technologies like Kubernetes emerged. The large overlap of actors that used to invest into OpenStack and now invest in Kubernetes indicates that newer solutions displace outdated *products*, but not the *community* of contributors engaged in this particular market segment. Collaboration is more important than the specific technology it produces, since products are comparatively short-lived. The resulting rather aggressive creative destruction showcases shortens innovation cycles and reduces large up-front investments in technologies that later fail to reach diffusion in the market.

Release early, release often describes a community norm of publishing intermediate results as early as possible, inviting feedback and contributions from other interested parties. It is related to rapid prototyping, but survives the exploratory steps and is also applied once a solution has gathered interest and a community begins to form around it. By enabling users of the solution to get involved in collaborative development early, the resulting feedback cycle is "an order of magnitude faster than most commercial software projects".[154] Since all contributions are almost immediately published in the product source code, all inventions embedded in them become prior art, and the original inventor is difficult to identify in the collaborative development process. Additionally, each individual contribution usually only embodies only a small, incremental inventive step. FOSS communities collaboratively invent, but are almost never able to acquire patents on their inventions even if they wanted to. This contributes to the lack of perceived utility of the patent system as a whole in the wider open source community, and to the reputation of the *OIN*, a royalty free cross-licensing network that covers core FOSS technologies in it's license agreement and has been joined by about 3000 FOSS participating entities.[9] Contributors exhibit an expectation of *shared stewardship* of the results of the collaborative process that extends to copyright, inventions, but also to the communities as entities. Additionally, because of the described dynamic of the constant generation of prior art, rights holders of patent claims that read on inventions originating in the wider FOSS community that do not cross-license their inventions under royalty-free terms or through OIN are almost exclusively non-participating or non-

[9] https://www.openinventionnetwork.com/ (accessed 03/02/2024)

practicing entities, further reducing their reputation and that of patents on computer-implemented inventions in general.

Voluntary participation in FOSS activities is fundamental to the understanding of the self-correction capabilities of governance in communities. Every individual or organizational participant contributes their resources voluntarily to the overall FOSS process. While this concept is well-understood and to an extent researched with regards to the motivation of individual volunteer developers [80], it also applies to businesses investing in community projects. There is no obligation to contribute, so organizations must have an inherent business reason to participate in general, and must be convinced that engaging in a particular community is the best alternative available to them. Because of that, communities are constantly pressured to develop governance principles that embody the *open source way* [155], and match the expectations of their existing and potential contributors. Once engaged in a particular community, participants face a voice-or-exit dilemma of constantly evaluating their choice to participate, and are always free to leave if they are not interested anymore.[65] This behavioral corrective combined with overall strong fluctuation of contributors ensures that governance within communities is self-healing, converges towards *open source culture*, and that communities appear, grow, shrink and disappear depending on how well they serve their purpose, and without necessitating public investment or interference. In this environment, it is highly unlikely that actors that violate community governance norms regarding shared stewardship or other aspects of collaboration will be successful.

The **upstream/downstream** model of collaboration within the wider FOSS community uses the mental image of a large river that collects the water from many smaller and smaller tributaries (the communities) and delivers it to the ocean (the users). To create products of the complexity of a whole Linux distribution like Debian, or the standard package index of a major programming language like Python[10], requires a coordinated effort of potentially thousands of individual communities. Product improvements originate in the communities and then are integrated "down the stream" by more and more complex aggregate products. Feedback like bug reports and requests for improvement, but also patches meant for

[10] https://pypi.org/ (accessed 03/02/2024)

integration into the upstream projects are generated closer to the users, and then travels "up the stream" to be eventually integrated by the community where the solution originates. Aiming for maximum reuse of code changes, the wider community developed a strong preference for *working upstream*, as in making an effort to have changes integrated as close to the original solution as possible.[155] FOSS licenses are applied to all products in the upstream/downstream network and facilitate the frequent integration and redistribution of aggregate products. Since there are potentially thousands of communities involved, the essential terms of all FOSS licenses are codified and always embed the "four freedoms of software", namely to use, study, modify and redistribute it.[140] Similarly, the wider FOSS community applies basic normalized processes of how bug reports and patches are disseminated upstream. Linux distributors for example participate routinely in providing smaller patches and reporting issues to a multitude of upstream communities with minimal friction. Tools like version control systems and issue trackers are used as de-facto standards that model a common workflow. More recently, Github has shaped the upstream/downstream model with their (proprietary) pull request workflow that has been very widely adopted. A complex network of many upstream and downstream participants following their own self-interest would be prone to anti-commons situations in which the cost of negotiations would grow to be prohibitive to collaboration.[62] Therefore the licenses applied by FOSS communities are *non-negotiable*. The upstream communities offer the use of their product under a specific license, and the downstream users accept that offer by using the software, or decide not to use it.[114] Similarly, but not to the same extend, behavioral norms like open access, transparency or meritocracy are non-negotiable in the wider FOSS community. The result are negligible transaction cost of participation in the upstream/downstream network. Standardized governance norms are beginning to emerge in the ICT industry and are facilitated by trade associations like the Linux Foundation or the Eclipse Foundation. An important role of these umbrella organizations is to establish common governance norms that facilitate collaboration in the wider open source community.

The weaknesses of FOSS identified in the interviews mirror the importance of the efficiency of the collaborative development processes.

FOSS currently lacks widely adopted **supply chain management** processes. The upstream/downstream model efficiently enables collaboration across the wider community, however it does not help businesses structure the relationship to their software suppliers. Unlike in other industry sectors, common attributes of deliverables like provenance of parts of the product or compliance with free software licenses are not commonly contracted. Out of the overall responsibility for license compliance of the final product, vendors need to evaluate the complete software stacks of their products, even if major elements of those have been provided by otherwise trusted suppliers. The well-established norms of collaboration in the wider FOSS community have not yet been completely translated to best practices or de-facto standards in the ICT sector, leading to inefficiencies, especially to high cost of maintaining license compliance and compatibility, and to barriers to the adoption of free software in commercial products. The OpenChain[11] project is a recent attempt to specify supply chain requirements. Long-term maintenance of FOSS products that matches the life cycle of durable industrial products is also difficult to establish.

Maintaining license compliance is a precondition for shipping products that contain FOSS or a mix of FOSS and proprietary code. Through legal instruments like injunctions, contributors that hold copyright on the product may stop businesses from selling their products if the vendor violates the obligations resulting from the redistribution of FOSS code. There is however a perceived uncertainty and arbitrariness in how free software licenses are enforced. License defense and litigation support is offered by community representatives like the Software Freedom Law Center (SFLC).[12] Cases also exist where individual copyright holders litigate against FOSS business users, sometimes for their personal gain.[95] Occasionally, businesses have even been recommended to avoid the family of GPL licenses altogether, which would unnecessarily lead to one third of the existing body of FOSS solutions not being used. This uncertainty has been addressed by the FSF in their “principles of community-oriented GPL enforcement” [58], with the “Linux Kernel Community Enforcement Statement” [78], and others. GPL-3.0 affords license violators more time

[11] https://www.openchainproject.org/ (accessed 03/02/2024)

[12] https://www.softwarefreedom.org/services/ (accessed 03/02/2024)

to correct mistakes before litigation can begin. However, most existing software licensed under the GPL still uses version 2. Some influential vendors releasing code under GPL-2.0 have begun to publicly commit to applying the GPL-3.0 “healing clause” to their products.[121] These initiatives indicate that managing license compliance is not yet as normal and standardized as it could be.

Meritocracy is a key tenet of governance in FOSS communities. Every participant, no matter if it is an individual volunteer, a representative of an organization, or an organizational entity itself, earns merit based on their concrete contributions to the community product. Contributions are however not all valued the same, in that one hour or one Euro invested would create equal merit. Improvements of the core product of the community are usually valued more than translations to rarely spoken languages or administrative support, even if they are just as important to the societal aspirations of FOSS communities. A Linux kernel developer gains vastly superior merit compared to a documentation author. On one hand, this is a direct expression of the auto-organization in communities. On the other hand, FOSS meritocracy discounts the importance of other stakeholders that represent interests in the community product but do not participate directly. This is partly intended – communities regularly try to focus their governance on those directly participating, trying to keep politics or “mere talkers” from interfering with the process. For regulators, representatives of civil society interests and other outside stakeholders, it may appear as if the FOSS communities evade responsibility for the externalities caused by the products they create. Even FOSS user groups happen to report a lack of interest by communities in their feedback. By building up loyalty to and standing within their community over time, an insider culture may develop where being a part of the community becomes an aim in itself. Contravening the idea of open governance, this erects barriers to entry for new contributors or previously not present stakeholders. Meritocracy in FOSS seems to be a double-edged sword.

The weaknesses of the FOSS ecosystem identified in the interviews represent risks from participating in the development process and from using FOSS in proprietary products, born out of a lack of supply chain management, compliance and governance standards. There is analytical bias in this study towards aspects of standardization since the interview candidates have been selected for their knowledge in this field. The results however match current trends in the ICT sector, especially efforts to mitigate *manageable risk* through the specification of supply chain, compliance and governance norms. While there have been improvements mostly from consorted industrial initiatives in recent years, at the moment businesses still cannot insure the *residual risk* of using FOSS. This indicates that the identified weaknesses are relevant and important for the continued success of the wider open source community.

4.5.3 Strengths and weaknesses of SSO

The strengths of SSO emphasized as most relevant with regard to this study during the interviews are the mature formal standardization processes, a powerful reputation in a large stakeholder network, reference and IPR frameworks, IPR, signaling of market opportunities risk of investment and value-added SSO services. Weaknesses are organizational inertia, on powerful stakeholders, an ecosystem where responsibilities are allocated based on historical developments, under-defined IPR frameworks and reliance on outdated revenue streams by some SSO.

Through a long history of setting formal standards SSO have evolved to be connected platforms with well-established tools and processes for formal standardization. The **formal standardization process** allows every legal entity to participate in the consensus-based development, documentation and approval of the recognized state of the art, while ensuring that all relevant stakeholders and the public are informed, consulted and invited to participate. Some SSO are obliged to consider social, environmental and public responsibilities during the formal standardization process, thus SSO are a great tool for the transfer and diffusion of technical solutions into the market, national and international economic policy implementation, market regulation, creation of legal certainty and implementation of security and safety measures.

The impact of standards development builds upon the **powerful reputation in a large stakeholder network** that SSO established throughout their history. The internationally integrated SSO ecosystem connects industry with government and research institutions. Participating in this network amplifies the reach individual actors can achieve in the market.

By successfully including a technical solution into a formal standard it changes into an accepted state of the art that more easily finds adoption among national and international implementers and business partners, significantly boosting diffusion in the market. Some SSO are accredited by governments or in other ways recognized to represent the interests of the countries economy and enterprises on the international stage. Participants in national standardization have the chance to gain global visibility if standards achieve recognition at ISO. Governments can influence formal standardization as an instrument to implement economic and trade policy.

The necessary cooperation between competitors in a market segment that is required for standards development raises concerns of collusion and anti-competitive behavior. An inherent conflict also exists between sharing technical solutions and the ability of participants to create a relevant IPR portfolio. SSO provide a platform to manage these concerns through **widely accepted terms of reference and IPR frameworks**. These frameworks help to attract innovative technology developers by providing policies regarding the disclosure and licensing terms of IPR covered by the standard, especially SEP. Licensing policies with regard to SEP are designed to give inventors the prospect of a return on investment on their research and development efforts, while still making these proprietary technologies available for standards development. Frameworks for disclosure and FRAND licensing commitments reduce uncertainty and maintain competition in the market. IPR policies ensure access to SEP, while leaving the concrete terms of later technology licensing to market-specific agreements. The inclusion of patented technology in formal standards is considered necessary for early standardization approaches in research and development heavy market segments and especially enable business models that rely on inventing and licensing IPR while leaving manufacturing to other parties.

The work program of SSOs usually follows long term roadmaps based on mission statements and standards development strategies. By connecting the activities of technology development and mass-market production, SSO **signal market opportunities** and facilitate research and development as well as manufacturing centered business models. The formal process of setting a standard may last between one and five years, which provides the necessary lead time to build manufacturing capacity and ensure standards compliance. For market segments with shorter innovation cycles, some specification instruments of SSO like DIN-SPEC[13] or the CEN Workshop Agreement (CWA)[14] reduce this process to less than a year by creating pre-standards rather than full standards. Released standards are systematically revised usually every three to five years. This allows stakeholders and the public to update, confirm or withdraw a standard and businesses to direct investments into implementing products. Technical committee members may have significant influence with regard to technical decisions. This increases predictability of standards development and acts as an incentive for market actors to participate in standardization, especially in market segments with high cost of change.

Especially long-term standards development participants may benefit from **value-added SSO services**. Formal standardization provides a secondary market where IPR and information are strategically shared among participants. SSO offer match-making between parties interested in standards development, discovery of technical trends and indicators for market diffusion. Engaged participants enjoy a club advantage from early access to market-relevant information and the opportunity to influence the resulting standards. This may lead to a competitive advantage especially for large enterprises engaged in a multitude of standardization activities.

The weaknesses of SSO identified in the interviews mirror the struggle with the faster pace of innovation caused by digitalization. This is true especially for technological fields with low costs of change, for example computer software, that are characterized by rapid technology changes and volatile trends. The formalized processes of SSO are at times too slow and too inflexible to adequately meet those market needs in time. The standardization process of some SSO is required to consider all stakeholder interests. This not only slows down the process, it also reduces the attractiveness of formal standardization for businesses that are under competitive time-to-market pressure. Industry actors may be incentivized to rather set de-facto standards to circumvent the formalized process. This makes it difficult for SSO to release market relevant formal standards in time.

[13] https://www.din.de/en/about-standards/din-spec-en (accessed 03/02/2024)
[14] https://boss.cen.eu/developingdeliverables/CWA/Pages/ (accessed 03/02/2024)

Interviewees mentioned a "standardization habitus" sometimes exhibited by employees of SSO. **Organizational inertia** manifests itself in an affinity for formal processes and little passion and vision for new approaches, new ideas and new topics. The staff of SSO focuses on administrative tasks and is usually far removed from innovation in the industry. As all institutions, SSO developed their own interests that may deviate from their original purpose, like maintaining budget and staff size or ensuring long-term employment safety. The fast paced innovation in the ICT sector may be in conflict with those goals. This is an issue that SSO like DIN and ISO have identified and are trying to mitigate jointly by following new approaches for attracting new experts or modernizing future technical committee processes.[70] CEN and CENELEC are investigating machine-readable publication and modern collaboration methods with the Digital Transformation Initiative.[50]

SSO depend on **powerful stakeholders** among the participating and member organizations for their budget and for developing standards relevant to the market. Especially large enterprises may use their influence to maintain control over standards that are important to their business interests. This limits the ability of SSOs to react to market signals and to adapt to new technical developments. Released standards may not be those most wanted by the market, or the standardized technical solution is watered down during consensus finding. Especially in highly concentrated markets with a small number of actors like the telecommunications sector, the influence of powerful member companies on SSO governance may undermine the covenants in terms of reference and IPR frameworks against collusion and anti-competitive behavior.

The distribution of responsibilities between the various national, regional, sector-specific and international SSOs developed against a background of strong nation states and a primarily industrial economy. SSOs form an **ecosystem where responsibilities are allocated based on historical developments**. National standards bodies sometimes develop competing standards in a "race to the moon" to claim relevance for new technological areas. This battle for relevance ignores that technical solutions today compete in a global digitized market. What role national standards bodies or even European SSO willplay within the global standardization ecosystem is a key question for their future relevance. This issue is more related to digitalization and improved methods of collaboration than to the relationship with the wider open source community.

While the IPR frameworks of SSO are mostly well-established and accepted and mainly considered a strength, the observed **under-definition of IPR frameworks** in some aspects has been identified as problematic. IPR frameworks establish common expectations and provide conflict resolution mechanisms crucial to standards development. To reduce uncertainty of legal risk, IPR frameworks are expected to provide clarity about the obligations and commitments of participants when the cooperation on standards development begins. Some SSO accept voluntary disclosures of SEP based on goodwill, which provides little safety against undeclared SEP. Some refer to FRAND, which in reality is not clearly defined. In most cases, the concrete agreement that the FRAND commitment promises is left to negotiation between competitors. Multiple interviewees referred to cases where technology users refused to license SEP, or where SEP holders refused to license to their direct competitors. Since the licensing agreements are usually confidential, participants cannot verify that the terms offered to them are in fact fair and non-discriminating. This acts as a barrier to participation in standards development, to market adoption and negatively affects the reputation of the respective SSO, especially when compared to the absence of negotiation when engaging in FOSS development. Royalty-free licensing of SEP can be considered a subset of FRAND, however it is only applicable to FOSS distribution if no explicit licensing negotiation or registration is required. In reality IPR policies of SSO that require any form of negotiation or license management are often generally avoided as a cultural mismatch or a distribution risk by the wider open source community.

Some SSO rely on **outdated revenue streams** that are in conflict to the general trends towards closer collaboration as well as openness and transparency. For example, some SSO that are set up as a combination of standards-development platform and standards publishing business continue to rely on revenue from selling access to standards for a fee. While such revenue generation may still be viable today, it can be expected that selling content that is essentially community-developed and can be copied at negligible cost may conflict with the overall trends in the ICT sector.

4.6 SSO and FOSS communities - partners or competitors?

After structuring opportunities and threats in the ICT sector as common external influences that affect both SSO and FOSS communities and then breaking down their strengths and weaknesses separately, it is apparent that SSO and FOSS communities both complement each other and compete for relevance at the same time, but for different aspects of the functions they provide. When comparing their respective strengths, there is no clear winner:

SSO possess a brand that promises tested, proven processes of standards development and a long-term roadmap providing stability to industry, research and policy maker participants. By defining and documenting the state of the art of technical innovation, they facilitate the interaction of a wide range of stakeholders and act as a bridge between industry and regulation. The wider FOSS community develops technical innovations at a fast pace, steers investments effectively through eliminating failing approaches early in the process and practices evolutionary product development with early and regular releases. Contributions are allocated self-healingly and efficiently through voluntary participation. Foundations act as the umbrella organizations that facilitate the upstream/downstream model. This opens the possibility for SSO to create standards based on "code first" FOSS solutions, similar to how the OpenDocument format was standardized [102]. This combination of a FOSS implementation with late formal standardization was repeatedly mentioned in the interviews and would apply the early competitive selection of FOSS processes to the selection of technical solutions for standardization. Even though this is not yet a common approach, it could improve the competitive selection of technical solutions for formal standardization. Since the continuous non-differentiating cooperation model reduces the need for specifications for interoperability reasons, formal standardization needs to be useful for other reasons, for example for quality management or compliance with safety standards or export regulations.

The most significant determinant identified in this study for which model is most efficient is *cost of change*. FOSS communities flourish in an environment where changes and corrections to products and specifications can be made almost instantaneously. The kernel development community integrates changes into Linux at an average rate of 8.5 patches per hour [35]. The cost of each individual change in this environment is almost negligible. Mistakes can be fixed easily and quickly, reducing the usefulness of specifications that would also be hard to keep up-to-date at this pace. On the other end, changes to the protocols used for wireless network communication potentially require hardware updates and would be costly and time-consuming. These extremes span a continuum where the *cost of change* decreases and the *pace of innovation* increases. Somewhere on this continuum is a point where the cost of change is the same for SSO and FOSS communities, dividing the field into two sectors where SSO or FOSS respectively are more efficient. The pace of innovation is usually inversely related to the cost of change, since software is easier deployed than new hardware. Commoditization influences the cost of change, making specification less relevant for products that become less costly to update. Based on this understanding, it is apparent that SSO and FOSS communities complement each other as processes for the management of technical innovation that are more or less efficient based on intrinsic attributes of the specific field of innovation.

Compared to the *strengths* of the FOSS community the *weaknesses* of SSO are mostly product development related and most apparent in environments with low cost of change.

Where communities allow numerous attempts at finding the best solution to a technical problem to fail early until a promising technology emerges, SSO sometimes "fail never" because of stakeholder inertia. In extreme cases they produce standards that are rarely or even never implemented. In a rather radical process of creative destruction, FOSS communities drive innovation in computing by being more efficient in the evolutionary selection of competing solutions. The wider FOSS community understands well that the release early - release often approach creates software faster and at higher quality, compared to the slow and thorough committee work SSO apply even in cases where it may not be necessary. Voluntary participation allocates contributions to where they are most useful and ensures that contributors are first and foremost enthusiastic about the software they create together. The communities attract eager inventors more than skilled administrators.

Individual volunteers contributing to FOSS usually self-identify with their tasks and work without direct remuneration. The communities they participate in may encourage, but not direct them to engage in standards development activities. Industry-driven communities may be sufficiently funded, but still not be motivated to invest into participating in standards development activities that in their perspective mostly benefits others. FOSS contributors that engage in SSO activities are unlikely to gain merit from that within their communities. Norms for participation in formal standards setting processes that expect them to be salaried representatives of a corporate entity which also funds their activities, membership fees and travel time pose a barrier of entry that is likely prohibitive towards the participation of FOSS contributors. To encourage them to participate, SSO require rules of participation and governance norms that reflect voluntary participation, and may need to align the sources of funding with the benefits from collaboration.

Some SSO are funded partly through the sale of access to standards. The trend towards openness and transparency raises expectations of open standards, at least in case they are mandated by regulators or in any other way preconditions for the entry into markets with any form of public investment. FOSS contributors will only collaborate in standards development if the resulting standards are freely available. SSO may be pressured to transition the share of their budget raised from selling access to standards to other sources of funding, like increased revenues from membership fees, professional services for their members and public funding. By providing services to facilitate their FOSS activities, SSO have the opportunity to offer a wider range of standards setting related services to their members from a single source, which may increase membership and open up new revenue streams.

FOSS communities are generally open for participation to contributors from all over the world. Sometimes it is difficult to identify which country a community is based in, and it does not matter much. The global setup matches the trend towards supra- and international regulation. In comparison, SSO may possibly be seen as narrow-minded especially in cases where they develop competing standards in different national standards bodies. The decentralized organization of the wider open source community incurs negligible transaction cost of a global participation.

This matches the mindset of inventors that rather work on developing products than deal with administrative processes.

The *strengths* of SSO compared to the *weaknesses* of FOSS focus on process and a narrow definition of meritocracy. Where FOSS communities struggle with establishing useful norms for supply chain management, SSO are not only able to provide stable processes and documentation that shape the supply chain for the industry and how regulators influence it, they even set explicit formal standards for it. Unlike FOSS, SSO are present beyond the ICT sector, and streamline the supply chain at a global scale.

Where FOSS communities struggle with the complexity of maintaining compatibility and compliance with free software licenses in complex software stacks, SSO establish IPR policies that are common across sectors. While there may be disagreement over what the best IPR licensing models are, SSO enjoy a higher acceptance as IP arbitrators across a more diverse group of stakeholders, including regulators.

FOSS communities are attractive to contributors of code. Often they lack funding and personnel for other, non-technical tasks like community management, marketing or political representation, making it difficult especially for smaller initiatives outside of influential groups to become viable. In particular if these projects have been initiated by some of their member companies, SSO could host and support them, offering similar services to FOSS umbrella organizations.

Where FOSS contributors attain merit based on a narrow definition of product-focused contribution, SSO provide well-accepted multi-stakeholder platforms and routinely involve civil society initiatives, environmental groups and other interested parties. The general acceptance of the balanced SSO process reinforces their brand as public benefit oriented stewards of technical innovation, which in turn secures sufficient funding and political support.

SSO represent well-established instruments for directing industry investments and policy making. Some FOSS communities are not yet prepared to take on similar responsibilities. Industry-driven umbrella organizations like the Linux Foundation are steering the community to adopt behavioral norms that enable them to engage with a wider range of stakeholders.

Even relating the *weaknesses* of SSO with the *weaknesses* of FOSS delivers interesting results. FOSS communities reward product contributions in their fast-paced meritocracy, while SSO apply a well-defined, slow and thorough process to their committee work. Innovations that do not fit into either template but still benefit from standardization pose a challenge to both camps. Especially large-scale ground breaking research that requires significant upfront investments like the development of pharmaceuticals or mobile communication protocols and hardware fit neither FOSS nor SSO, and is usually performed by industry consortia or competing large enterprises. Neither SSO nor FOSS are well-equipped to deal with the resulting thickets of SEP, especially against the general trend towards more openness and transparency. As a result, industry sectors where SEP are an inherent element of invention exhibit difficulties of adopting FOSS. Suppliers in these markets tend to participate mostly in sector-specific SSO like ETSI where they can exert strong influence over the developed standards.

Globalization challenges the role of national standards bodies and the established order of SSO in general, while FOSS still lacks well-working norms of supply chain management. Outside of FOSS, standards are still not developed at a global scale, and are still considered instruments of industrial policy by regulators, even though the macroeconomic benefit would be maximized by the widest possible adoption of a standard. To avoid the history of different railway gauges and power socket designs from repeating, standards should be developed in global collaboration, which is the case for many national standards bodies under the umbrella of ISO, but not necessarily for sector-specific SSO. FOSS still struggles with this responsibility, and some SSO are not yet ready to give up part of their territorial authority.

The IPR licensing policies of SSO have originally developed against the backdrop of an industrial society. They where meant to manage access to specifications of physical goods with long product cycles where the cost of change is high and the effort of specification or even publishing standards as printed books is comparatively low and justified. FOSS communities developed tools for immediate development of specifications and implementations in public, and are still working out how to maintain license compatibility and compliance in complex software stacks. These approaches barely overlap. As a result there is no common platform, for example, to establish what the requirements for open standards are, or the exact meaning of FRAND, or the right balance between enforcing standards with mandates and making them freely available, and many other fundamental questions about the future of standardization.

The "open source way" enables the development of new models of collaboration, especially continuous non-differentiating cooperation (see 4.3.3). The co-existence of SSO and FOSS communities affords participants in technical standardization a choice of which of their complementary models best fits their particular environment. Many of the arguments produced by FOSS proponents emphasize the political nature of the free software movement. The majority of communities today however understand it as their mission to create software and establish industry standards. They apply the FOSS model because they believe it is the better way to achieve that goal. This pragmatic approach should not be misunderstood as indifference towards the ethical underpinnings of software freedom. Most contributors are consciously aware that by creating FOSS, they are also being virtuous by benefiting the common good. The realization that contributing to free software means being enjoyably productive and doing the right thing at the same time is key to understanding the success of FOSS. Participants expect reciprocal behavior and shared stewardship of the results of collaboration as well as transparency and openness in the process. Models of collaboration between SSO and FOSS need to reflect that reality.

4.7 Managerial and policy recommendations

The goal of this study is to compare SSO and FOSS processes and activities that create a standardizing effect and discuss potential implications for FOSS communities, SSO, regulators and enterprises. After analyzing the strengths and weaknesses in context of the ICT sector specific threats and opportunities, our study supports a number of conclusions about the interaction between SSO and FOSS from a product, process and societal perspective.

The product view is that formal standards are products that specify the details of how a technical solution should be implemented, while FOSS creates products that implement these solutions. SSO are better at developing formal standards, while FOSS communities have an advantage at collaboratively developing non-differentiating implementations. If there is demand for both specifications and implementations, SSO and FOSS can and already do successfully complement each other.

SSOs implement a top-down, specification-first process, while FOSS communities implement a decentralized, code-first process. For every particular product development activity, participants need to choose one or the other approach. Depending on which process design is more applicable to the desired outcome, actors will choose to participate in SSO or FOSS. For endeavors that involve the development of multiple products, like a formal standard and a reference implementation, different approaches can be combined, and the results may cross over between them. Regarding process, SSO and FOSS are competitors.

At the societal level, SSO report to regulators and are used as a policy instrument to define safety requirements, competitiveness as well as antitrust regulation and to implement industrial policy goals in general. FOSS communities operate based on voluntary participation and an authority that rests within the communities. Regulators have a choice to influence, support or provide guidance to both SSO and the wider FOSS community to drive innovation and competitiveness and to enforce policy goals. From a societal perspective, SSO and FOSS are independent and complementary standardization instruments available to policy makers.

Implications for FOSS communities: The wider open source community will benefit from applying a principled definition to what makes a contribution and from treating all contributors as equals, no matter if they are individual volunteers, businesses, research organizations or governmental agencies. Historical prejudices sometimes shared in the “hacker community” that free software is the prerogative of civil society and hackers are not justified considering that today’s community composition includes a majority of corporate contributors. Global FOSS collaboration offers a unique opportunity to build bridges between the interests of individual, civil society, enterprises and the state, as long as they are willing to contribute to the free software commons. The wider community can work towards realizing this potential by actively influencing the quality of the collaboration process as perceived by the different types of participants through community management and setting governance norms, while maintaining and reinforcing the principles of free software like meritocracy, transparency, non-discrimination, shared stewardship and open collaboration. By identifying and lobbying for software freedom as the overarching goal, the FOSS community can build the foundations for a successful integration with regulators and societal institutions like SSO. Only standards that are accompanied by specifications are likely to be referenced in legislation, so communities have a choice to produce the specifications themselves, or to collaborate with SSO and benefit from their reach, brand, and network for that purpose. Communities should expect to be held to higher standards regarding social responsibility in the future, and should consider this a sign of their success in contributing to the common good.

Implications for SSO: Standard developers could invest into gaining deeper knowledge of the mechanics and principles of FOSS collaboration to understand the strengths and limitations of the open source way. Knowledge of the applicability of FOSS to software and software/hardware interface innovations will help to delineate activities in higher cost of change areas from the domain of FOSS communities. SSO can confidently position themselves in technology areas where high cost of change and a slow pace of innovation make diligent up-front specification worthwhile. When engaging with FOSS communities, it should be considered important to understand the behavioral norms that have solidified in the collaborative zone. SSO should be especially wary with violating the expectation of shared stewardship by supporting or facilitating the appropriation of collaboratively developed work by private enterprises through SEP or other IPR that may become standard essential, like trademark licensing policies. SSO may decide to support and encourage participation by FOSS contributors even if they do not create direct revenue from their participation. Membership fees, requirements for nominations to committees and other barriers to participation could be waived since they not only pose a barrier of entry, they also undermine the acceptance of the SSO by violating meritocracy. Resistance can be expected from some SSO member companies against facilitating FOSS activities either out of a lack of understanding or because the results may negatively affect how participating in standards development aligns with their business interests.

Regarding their own business models, SSO should actively reconsider and prepare to move away from models that depend on selling information goods like copies of a standard for a fee. Price discrimination in the form of giving free copies of standards for research or FOSS use is already applied in some places and may continue to be an intermediate solution. In general, membership fees for participating enterprises are accepted in the wider FOSS community, while fees for the use of specifications or products are more controversial. SSO may consider to accept that the traditional approach to standards setting does not work well in the low cost of change/high pace of innovation software industry, and decide to instead shift some of their activities so that they become part of the wider open source community. By accepting the FOSS community as equals in the standards development field, SSO could become part of the meritocracy of the community itself and broaden their service portfolio by offering their members insight into standardization and FOSS activities related to a specific field of technology.

Implications for enterprises: To remain competitive, enterprises may need to focus their R&D spending towards features that differentiate their products in the eyes of the consumers. Enterprises pool investments into non-differentiating features with other collaborators or even competitors, reducing their share of the cost of provisioning the necessary functionality. This allows them to increase the share of research and development spending invested into what makes their own products unique under a policy referred to as "differentiate or collaborate". Participating in FOSS activities is a proven method for continuous non-differentiating cooperation collaboration (see 4.3.3). It is a complement, not a replacement for engaging in standards developing activities. To successfully participate, enterprises need to understand the different behavioral norms applied in the collaborative environment of FOSS compared to the competitive environment. The general application of non-negotiable licenses and governance norms means enterprises should refrain from attempts to appropriate the results of collaborative development processes through secondary restrictions to software freedom. Once it is accepted that FOSS products represent the non-differentiating state of the art, acquiring exclusive rights to functionality developed in the collaborative environment becomes meaningless. To be accepted as "good citizens" of the wider open source community, enterprises should continue to adapt to the behavioral and governance norms applied by the wider open source community when participating. The decisions to participate in the specification and the implementation of standards need to be made separately and considered complementary.

Implications for regulators: SSO and FOSS engage in a healthy competition from the process perspective which fosters technical innovation. For both to be considered alternative industrial policy instruments, regulators may choose to create a level playing field. This requires creating exchanges between the evolutionary selection process in FOSS and the formalization of SSO, but also may add additional obligations like working with multi-stakeholder platforms or adherence to minimal standards for governance norms that support the long-term viability of the FOSS development model. For this purpose, the role of FOSS umbrella organizations should be considered closer to that of SSO than the role of individual FOSS communities, which is also apparent in their membership structure as well as in the variety of projects hosted by the foundations. The public support of FOSS foundations could be raised to a level comparable to the support provided to SSO, especially if they commit to a charitable cause. The acceptance of the public interest in the contributions FOSS makes to the common good could justify the establishment of European FOSS development organizations either next to or integrated with existing SSO. Careful consideration needs to be applied to avoid disrupting the upstream/downstream model peer production process that is based on self-identification. This can be avoided by selectively awarding competitive, time-limited grants similar to current research funding by the EU. Governmental and regulator representatives should expect to be received as welcome contributors, but also to have to earn their merit in the communities like any other contributor. When developing policy measures aimed at fostering FOSS development, sector specific experiences may not be generally applicable. In particular the highly concentrated, regulated and politically influenced mobile communication sector may not be a useful yardstick for the development of general public FOSS policy. Experiences from a plurality of highly innovative technology areas like cloud-native computing, automotive platforms or programming languages that involve standards setting and implementation should be taken into account. Practices need to be developed that reflect the trend towards openness and transparency. Exclusive third party rights to formal standards that are mandated or where compliance is a barrier to market entry will find less acceptance and may be considered by the public as inappropriate rent-seeking or invitations to morally hazardous behavior. Next to welfare losses from the lack of adoption of formal standards, public policy that facilitates SEP may undermine the competitiveness of local industry sectors by inviting outsiders to compete by participating in global collaboration on developing FOSS solutions. The availability of formal standards where compliance is mandatory or practically required as open standards would eliminate this possibility.

The assumption that the state should avoid displacing the economic activities of private enterprises in the market does not generally hold for FOSS activities. If the collaboratively created product is made available under a FOSS license as a public good with perfectly infinite supply, contributions by the state do not negatively affect the market or compete with specific private enterprises. Government agencies may decide to participate in FOSS activities to satisfy their own needs in collaboration with community and enterprise contributors without negatively affecting competition, which supports the arguments made by the PMPC campaign [55]. Public contributions to FOSS should be considered pro-competitive as they enable all enterprises to refocus their investment on differentiating product features. The argument that all information goods produced by the state should be released under FOSS licenses since they belong to the public should be investigated earnestly. Establishing an agency to foster and support FOSS development at the EU level or tasking existing EU SSO or national standards bodies with that responsibility will help to leverage the innovativeness of the wider open source community.

4.8 Summary

To answer the question of whether SSO and the FOSS communities are partners or competitors, our study identifies significant changes induced into the ICT sector by digitalization (see 4.5.1). Improved methods of collaboration, a global trend towards openness and transparency, the shift in the role of the modern state to a regulator and a transition from national to supra- and international collaboration and regulation present both opportunities or threats to ICT market participants.

Against these changes, we analyze standardization as a *path* that leads from ideation to industry-wide diffusion through specification and implementation (see 4.3.1). To answer perceived standardization needs caused by market pull or regulatory push, we introduce the concept of standardization instruments that cause the transition to a widely adopted standard by exerting standardizing effects. Based on that, we introduce a phase model of standardization that is able to describe early, late and parallel standardization approaches. By deciding to participate in standards setting, FOSS development or both, participants elect standardization instruments that promise to efficiently lead to market diffusion. We believe this standardization phase model explains more types of standards development activities compared to the document-centered ISO definition or institutional models that assume standardization is what SSOs do. In particular, it describes the processes of both SSOs and the wider open source community.

In the SSO ecosystem, we find that the reputation of individual SSOs in the ICT sector depends on relevance more than on formal recognition (see 4.5.3). The importance of interoperability changes as new technologies may now be incubated as FOSS before being standardized, leading both to higher quality standards and potentially to single joint implementations available as public goods, decreasing the importance of data format or communication protocol specifications. With global collaboration methods available, especially national standards bodies now compete more closely for relevance, which opens up both opportunities for a more efficient distribution of responsibilities and a challenge to retain mind share and to restructure accordingly.

To better illustrate the FOSS ecosystem, we introduce new or more precise definitions for the concepts of community, the wider open source community, umbrella organizations, contributors, contributions, FOSS licenses and the upstream/downstream model (see 4.5.2). We believe that these concepts are intuitively well-understood by participants, but so far not present in academic literature in a coherent and interconnected way. We introduce the concept of continuous non-differentiating cooperation that partially replaces pre-competitive cooperation, and identify that industry participants apply a principle to either differentiate or collaborate. Voluntary participation, indicated by the competition for contributors and the looming threat of forks, acts as a corrective force that cause community governance norms to converge towards the expectations of the wider open source community. Our study reiterates that FOSS licensing minimizes transaction costs by eliminating negotiations, thus preventing the numerous FOSS licensing relationships from causing an anticommons situation.

When assessing the relationship between SSO and FOSS, the relevant standards need to contain software components or interfaces between hard- and software components (see 4.6). Standards exclusively covering physical goods are not implemented as FOSS. Our study identifies cost of change and pace of innovation, which are usually inversely correlated, as factors that influence the decision to focus on specification or implementation first. The results indicate that SSO and FOSS communities compete for relevance as alternative process choices based on their suitability to address specific needs for standardization, and that some topics where both SSO and FOSS processes show weaknesses, like early-investment heavy research and development efforts not suitable to open collaboration, are currently not efficiently served by either process. Finally, the results show that the roles of FOSS umbrella organizations and SSO exhibit signs of convergence, with both influencing the governance and IPR frameworks of innovation and the development of standards and acting as platforms for consensus building.

Overall, our study describes a utilitarian approach to standardization where a perceived need for a technical standard is satisfied by the diffusion of a technical solution in the market through standardizing effects caused by the application of standardization instruments. This approach signals that standardization serves a purpose instead of being a goal in itself. SSO and the wider open source community both cause standardizing effects. By creating a level playing field where SSO and the wider open source community can collaborate in technical standardization and also compete as standardization instruments based on voluntary participation, innovators and policy makers can ensure that the most applicable process is chosen for the development of a standard. SSO and FOSS are competitors and complements at the same time. Combined they contribute to realizing the opportunities and help overcome the threats that result from globalization and digitalization. By recognizing the contributions of both SSO and the wider open source community to the common good, society has the opportunity to build the foundations for modern institutions that shape and facilitate technical innovation.

4.9 Thank you notes

This study would not have been possible without the help of many motivated individuals from academia, industry, the standards development community and the FOSS ecosystem. In the spirit of open collaboration, the authors invited comments and critique as the study progressed. We would like to thank the fellows of the OpenForum Academy, the members of the Legal Network and the representatives of IEEE-SA, DIN, the Linux Foundation and the Open Source Initiative and of course our fellow academics for their incredibly valuable feedback. All errors and mistakes inevitably remain our own fault.

The interviewees who participated in this research each have a long standing professional background and are long term contributors to their respective roles and organizations. We would like to thank Alpesh Shah (IEEE-SA), Dave Neary (Red Hat), David Faure (KDE Community and OASIS), De-Won Cho (DIN e.V.), Dirk Weiler (Nokia and ETSI), Mike Schinagl (Document Foundation and FSFE) and Thorsten Behrens (LibreOffice) for participating in our study.

For further research the recorded expert interviews are archived and can be obtained at the chair of innovation economics at the Technical University of Berlin.

Chapter 5

The economics of open source

This article was originally published as: Mirko Boehm. “Economics of Open Source”. In: *Open Source Law, Policy and Practice.* Oxford University Press, Oct. 2022 (pages 298–328). Reproduced by permission of Oxford University Press. For permission to re-use this material, please visit https://global.oup.com/academic/rights.

The relationship between Open Source and economics is fundamental since the collaborative creation of software and its utilisation are economic activities. There is a value that businesses can generate with products that include or consist solely of Open Source, and a potential cost saving in its use. The work of the wider open source community is coordinated and software is created by different elements of the Open Source ecosystem. It is integrated as an intermediate or final product into consumer applications that deliver concrete, useful functionality. Open Source is unique in that it is simultaneously state-of-the-art technology, a commodity, and a public good. Open Source communities are social groups of individual and organisational contributors that participate voluntarily in the production of public information goods.

Communities are able to resolve issues without the coordination provided by a central authority like the state. Overall, the wider Open Source community contributes positively to the common good. This shapes how Open Source collaboration relates to market competition and the value propositions of businesses. Today, collaboration spans both unpaid or volunteer participation and industry contributions that are made in the course of employment or under the guidance of a commercial sponsor. This chapter develops a basic taxonomy based on a combination of the revenue model, the type of good, and the differentiating aspects of Open Source-based products. It positions Open Source in its relation to economics and discusses the different behavioural norms like reciprocity and fairness that participants apply to the social transactions of Open Source collaboration, as well as the impact of Open Source on the technology stock of society.

In recent years, Open Source has gone through a remarkable transition from an exotic pastime of idealists into a mainstay of software engineering practice. Today's communities are a diverse mix of individual volunteers and industry contributors who collaborate on software development, while simultaneously competing intensively in business. Open Source is used pervasively throughout the ICT sector. Such pervasive adoption is inevitably driven by sound economic arguments that appeal to many actors with differing motivations and self-interest, indicating that there is a theoretical foundation for the mass appeal of Open Source collaboration and for the reconciliation of self-interest or business rationale with collaboration on the development of Open Source technologies

5.1 Introduction: Open source, law, politics and economics

Open Source itself is first and foremost source code and therefore software. This is the way many people see it— as an amazing pool of free software modules to select and build upon. However, that is not the whole story. Free licensing of source code combined with collaborative and accessible development processes create a relationship between Open Source and a cross-section of society. Richard Stallman, the founder of the FSF, insisted that 'free software is a social movement'. [140] Software is a form of technology with wide-ranging social implications. In a society which has digitised, it impacts a diverse range of issues, including civil liberties and human rights, access to the means of production of goods and services, methods of collaboration, and many others. Three of these relationships that have attracted particular attention in recent years are those between Open Source and law; politics, and economics.

The interaction between Open Source and law manifests itself in Open Source licensing and compliance. Source code that is created by humans is covered by copyright.[135] In most cases, explicit permission from the copyright owners is required to use the code due to the application of copyright. Open Source licences are the mechanism by which authors give permission to third parties to use their work. These licences form the basis of the legal relationship between the authors and the users. Disagreements between the owner/licensor and licensees are resolved by recourse to legal processes. Other legal frameworks that intersect with Open Source include the potential to own and infringe patents in the software in some jurisdictions.[6]. Chestek explores the use of trademarks to designate product origin.[30] Contributors sometimes appoint fiduciaries that represent their rights or enter into agreements with communities or businesses.[128]. Questions also arise around liability for code which is either negligently or maliciously constructed. Issues around rights and obligations around the development, distribution and use of Open Source today are dealt with primarily through private law.

How Open Source and politics relate is possibly more abstract, but already hinted at in Stallman's statement quoted earlier.

Open Source initially challenged how the software industry innovated, by shifting from a proprietary or closed to an open and collaborative model. As software has become increasingly important in many different sectors of the economy, through a process of digitalisation, the effects of Open Source innovation have become pervasive. In particular, the collaboration methods developed by the wider Open Source community have inspired related changes in business activity around software development and also inspired changes in areas not directly associated, such as open access in science, open data, or open hardware.[75] As part of this shift in how knowledge is transferred and monetised and how technical standards are developed and adopted, Open Source has offered new and alternative approaches to innovation.

Other changes go deeper and reflect technological development over the last two decades. Ubiquitous Internet access combined with Open Source enables previously unfeasible participatory forms of decision-making in society, alternative approaches to knowledge transfer, and opportunities for provisioning public ICT infrastructure, with reduced lock-in to specific technology providers. The gig economy has transformed the labour market and impacted employment patterns and worker mobility. Understanding of the societal changes caused by the Internet with resilient connectivity, convergence, and software freedom is still at an early stage, which is why this chapter focuses primarily on the *micro*-economic effects of Open Source.

The relationship between Open Source and economics is fundamental. Economics studies how our societies produce and trade goods and how individual actors make decisions when participating in that process. From an economic and, probably, a legal perspective, software is considered to be a good. When considering economics, it would traditionally need to be traded to be useful.

Development of software and in particular Open Source is an inherent part of the economy. In the case of Open Source its contributions to GDP may be difficult to measure since using it is generally not accompanied by a monetary exchange (although there is no prohibition on such it is not the norm).

Open Source is a software good with specific properties. In particular, it is available without significant restrictions and in unlimited quantities and allows every interested party to use and improve the existing source code, build upon it, and redistribute that modified version. This raises two key questions:

- What incentivises an individual to consume and to contribute to Open Source?; and
- What creates the balance between supply and demand so that the market is provided with the software that is needed in the necessary quality and quantities?

From a macro-economic perspective, a choice that affects us all is how society may react to the changes that Open Source imposes on it. They could be *rejected*, for example because they threaten jobs in established businesses. They could be *tolerated*, because the benefits of innovation outweigh the potential costs. Or Open Source could be *facilitated*, invested in, protected, and supported because of a belief that it is a beneficial pillar of the digital society. To make that choice, it is necessary to understand the overall impact of Open Source on society.

5.2 Why is free software free?

There is an unfortunate confusion based on the meaning of the term *free* in the English language. It means both free as in free of charge as well as free as in freedom or liberty. In the case of Open Source, it is free as in liberty, but generally also free of charge, licence fee, or royalty. To understand the economics of Open Source, we must also consider not only what makes Open Source free but also why our society embraces a model where important software technology is developed in an open, collaborative model. In short, *why* is free software free?

Software begins its life as the human-readable source code. Without the application of IP protection in the form of copyright, source code can be regarded as merely information. Information is produced, traded, and has value, making it a *good* in the economic sense. Information however has properties that make it special, in particular *intangible*. It is "costly to produce but cheap to reproduce"[133]: today, reproducing (making a copy of) information on a computer practically incurs no cost at all and, increasingly, all information relevant to human activity and existence is represented in digital form on computers.

When humans consume information, for example by reading source code or experiencing a piece of music, they convert it into knowledge, a process that cannot easily be reversed (some say information is difficult to dispossess). Information generally available on the Internet can be consumed by any interested user, making it *non-excludable*. Each user's experience will be mostly unaffected by the fact that others are consuming the same information at the same time, making it *non-rivalrous*. Products that both non-excludable and non-rivalrous are defined as public goods. Information without property protection is either secret or potentially available to everybody.

There is a dilemma in that it takes effort and creativity to produce valuable information while it is easy and cheap to reproduce it. This is well-understood, and one of the foundations of the Berne Convention, a pillar of international copyright law ratified by all developed countries in the world.[159] It posits that authors acquire copyright on their works as soon as they are 'fixed', and that others need explicit permission (a licence) from the copyright owner to use, reproduce, and distribute their works. By giving authors a legal instrument to manage who has access to their works, copyright provides the framework that makes intangible information goods tradeable.

In most traditional uses, copyright and its licensing are applied to restrict the number of available copies of a work. In a competitive environment, a good that is available with unlimited supply will converge on a price of zero. This fact incentivises rights holders to limit the number of copies available in the market. For example, reproductions of paintings may be limited and books printed in batches. Binding the information good to a medium for transport illustrates its intangible nature. The number of copies of the medium is restricted as a means to limit supply and to maintain a non-zero price. If the cost of creating another copy is very low, as for example with digital music streamed from the Internet, subscription models are an example of a tool to generate revenue based on the aggregated market demand.

Open Source moves a step forward in the application of copyright to digital goods. It applies the same concepts of authorship and licensing discussed so far and applies terms that make the software available in *unlimited supply*. All Open Source licences guarantee that users have the rights to use, study, modify, and redistribute the software. Since everybody and anybody can redistribute the code, any piece of Open Source is available to the general public once released and very difficult (if not impossible) to retract. That means a piece of software released under an Open Source licence begins life as a public good, and is made into a private good by the copyright acquired by the author, and then reverted to a public good again by the application of the Open Source licence.

However, since users have the right to use the code on the terms chosen by the author, these may contain obligations or restrictions. For example, attribution, as in naming the authors of the used Open Source building blocks in all reuse is a minimum requirement in these licences. The class of reciprocal, 'copyleft', Open Source licences require that users distribute their own modifications under the same terms, ensuring that the software remains Open Source even if modified.

The use of traditional copyright licensing in a way that enables sharing and user freedoms, effectively playing the IP right at its own game to revert its impact, is a 'stroke of genius' attributed to Richard Stallman. As a consequence of this action, all Open Source licences are anchored in the author's copyright and precluded the need for an Open Source-specific legal framework or 'lex Open Source'.

5.3 Software freedom and open collaboration

The basic concept of Open Source mixes two perspectives, that of free software as a product and that of open and collaborative processes for the development of software. A typical understanding is that an Open Source product is both free and also developed in an open, transparent collaborative process. These two perspectives are distinct.

Software is considered Open Source if it is made available under an accepted Open Source licence. Today's understanding of the formal requirements for code to be Open Source builds upon the 'four freedoms of software' as laid out by the FSF [54] and more generally the Open Source Definition [106] maintained by the Open Source Initiative. All Open Source licences give the users of the software at least four essential rights, namely to use, study, modify, and redistribute the software without discrimination between who uses the software and for what purpose. This definition does not prescribe in what way the software is created.

Open Source contributors may not consider their works to be products since they are not sold in the market. In the context of this article, the term 'product' is understood as a good made available for another's use. The requirement is that having been produced the good is made available, not that it is being sold at a price. Various goods are made available for free even though they are costly to produce, for example in state-provided free education.

Open Source may be created in a community process where interested parties may participate and contribute to its production based on the merit of their contributions. What if any requirements of this *openness* exist is still the subject of an ongoing debate. In most communities, it means that all contributors are welcome and should be treated respectfully and equally.

However, there are businesses that maintain control over an Open Source product or a commercial version of it, while accepting outside contributions. A company may act as a commercial sponsor to an Open

Source product, usually combining it with offering a complementary commercial version or support, such as Canonical sponsoring the Ubuntu operating system.

There are however FOSS products where one organisation maintains control over it while happily accepting outside contributions. Another form of development is the release of projects originally developed in-house under an Open Source licence, like Google's Kubernetes. Many projects today start as industry collaborations where businesses cooperate on the development of a foundational technology or an industry standard. Such projects are often set up at Open Source foundations.[128]

Some refer to this by saying 'there is more than one FOSS way'.[108] Since this Open Source governance is less standardised than Open Source licensing, it is assessed indirectly by measuring accountability, transparency, or the accessibility of the community decision-making processes.[21] Calling software Open Source conveys certain positive values on the code and may be used by businesses in marketing, whether accurately or otherwise, adding to the confusion.

The distinct merits of Open Source products and open collaboration are each economically relevant. Similar to inventions, Open Source products become part of the technology stock of society and influence the state of the art of products and production processes. Due to their free nature they may be the subject of both rapid adoption and ubiquity.

Even though Open Source is usually distributed freely, its functionality has an impact that has value and therefore impacts GDP. Open Source development *processes* enable efficiency gains that contribute to economic growth for example through improved interoperability or as a consequence of their free adoption may eliminate duplicated efforts.

5.4 Differentiate or collaborate!

According to the Maddison project database,[23] real GDP per capita in Germany increased more than twenty-fourfold between 1850 and the year 2000. Other industrialised countries show similar increases.

GDP per capita depends primarily on the technology stock applied to production since it does not increase if more people produce the same amount per person. It is plain to see that the dramatic increases in the standard of living experienced in this period depend heavily on technical innovation. While it is difficult to provide a clear definition of what innovation is,[104] its effects are real. It is the improvement of the technology stock available in an economy that leads to increases in real GDP per capita and lays the foundation to improve the general standard of living. In plain terms, it is the 'work smarter' aspect of economic growth.

Innovation is tied closely to competition.[1] Competition in a free market is a somewhat Darwinian concept that is supposed to keep businesses honest and aligned with consumer interests. Businesses that are out of touch with the needs of their consumers tend to fail and be replaced by better-performing competitors. Since most of the well-developed economies in the world are market economies, it is often assumed that competition is necessary for economic performance. Competition, however, comes at a cost. Schumpeter aptly describes one such cost as 'creative destruction'.[130] By introducing improved products and manufacturing methods, the value of earlier investments in outmoded products and now-redundant facilities are destroyed.

A further cost of competition are invention races. It is common in the startup culture of the ICT sector today for multiple new ventures to invest in the same trend or solution to a problem, only to drop out of the race once there is a clear winner. The others who did not win this race fail to deliver return to their investors. This issue is especially apparent in patent races, where the first inventor to be granted a patent wins, leaving the competitors in the dust with almost finished inventions that are now mostly worthless due to the monopolistic IP protection afforded to the first to register a patent. However, competitive markets provide high-quality products at low prices to consumers. The benefits clearly outweigh the costs. Competition appears to be a 'least-bad' approach, one that makes businesses work for the consumer at an acceptable cost. This is reminiscent of Churchill who described democracy as 'the worst form of government, except for all the others'.[143]

In every economy, regardless of the type of government, competitive and cooperative processes coexist. Competing businesses may cooperate on standards development. Public goods, like education, may be cooperatively provisioned by the state in a centralised fashion. Decentralised cooperative production has, however, historically represented a negligible segment of the economy.

Such collaboration has always existed in the form of neighbourly help or barn-raising.[1] It has generally been limited in scope by proximity and shared interest. Today, the Internet enables global collaboration on Open Source development and proximity is no longer geographically restrained, which in turn allows interested parties to collaborate globally.

The beginning of the Open Source ecosystem and the development of the Internet coincided. Together they triggered advances in collaboration techniques like wikis, issue trackers, and revision control systems that enabled widely dispersed groups to work together.

Open Source offers an alternative to market competition that enables participants to collaborate where they do not plan to differentiate.

Differentiation is businesses' understanding or belief of what product features convince consumers to choose their products over those of their competitors. While those differentiating features are developed in-house and usually kept proprietary, there is no business reason to invest in the development of non-differentiating functionality individually, duplicating efforts of competitors. For example, every computing device needs an operating system, which is a complex and crucial piece of software. Consumers, however, almost never interact directly with it and are usually indifferent as to which operating system their device uses. While in the 1990's it was still common that every printer manufacturer developed their own firmware, today almost all of them are based on Linux. The same logic applies to the foundational software stack in general that is used to build consumer-focusing applications.

It is assumed today that devices contain over 80 per cent Open Source software with the remainder being proprietary, differentiating code, which has been called the *Pareto Principle of Software*.[101]

This trend comes with a drawback. To be able to build competitive products, a business *must*, in addition to the use of its own differentiating code, use the available Open Source software stack of non-differentiating software to the fullest extent possible, since its competitors will do so, and otherwise undercut their cost. As Open Source reduces R&D cost, with such costs being shared across the creators of the code, this is factored

[1] http://amishamerica.com/what-happens-at-an-amish-barn-raising/ (accessed 03/02/2024)

into product prices today.

As a consequence of this 'differentiate or collaborate!' has become a mantra in today's ICT industry, as the guide to decision-making for competitive, differentiating product features.

5.5 Joint stewardship and governance

By integrating Open Source, producers acquire crucial functionality for their devices at the expense of a partial loss of control over the software functionality of their products. Much of Open Source is developed in an open innovation model as opposed to the traditional confidential corporate R&D models.[29] As a consequence of ongoing collaboration by the development community, incremental changes to the Open Source modules are routinely and continuously shared between the participants, usually in the form of commits to a version control system such as Git.[2] Users benefit from gaining access to the aggregated contributions of other participants, the value of which commonly outweighs the cost of their own contributions. For industry contributors, participating in Open Source development is primarily an approach to save costs and increase speed of innovation by pooling R&D resources in cooperation with other participants.

One consequence of this collaborative innovation approach is that the developed Open Source functionality is available to everybody, also including non-participating parties. It is also available to contributors' competitors as others cannot be restricted from using the same code and building upon it or modifying it. By attracting contributions from many diverse stakeholders, Open Source can be very innovative and represent the current state of the art in a specific field. At the same time, it is considered a commodity in that the functionality it provides loses its differentiating value and becomes generally available to the whole world.

The combination of innovativeness, commodity character, and being a public good makes Open Source rather unique. It drives the innovativeness of the wider Open Source community and explains why Open Source development contributes positively to the common good.

[2] `https://git-scm.com` (accessed 03/02/2024)

It is possible to use Open Source without ever participating in its development. Indeed, an overwhelming number of users fall into this category. Passive consumers, however, will be unable to influence the technical development of the software beyond making feature requests in communities. This exposes them to business continuity risks. As the development of the software continues, it may deteriorate in quality or deviate from the functionality needs of the passive consumer's application.

Businesses consuming Open Source therefore have a stake in the viability of the ongoing development of the Open Source components they use. By engaging in the community and investing in shared development efforts for that software, businesses ensure that the Open Source products they consume match their functionality and quality requirements and, importantly, that it will continue to be maintained and secure. Invested consumers of Open Source often become contributors to it in the long term. The process is a cycle where users frequently begin to participate in both the creation of the software as well as in the governance of the community that develops it as a way actively to manage their own business risk of growing dependence on the Open Source components they use.

5.6 Contributions, copyright and participation

Participants contribute code to Open Source projects which is the subject of copyright. The resulting releases of the packages usually contain contributions from many different contributors. Each release builds upon the earlier versions as a derivative work. Since the contributor base of Open Source projects fluctuates over time, the set of rights holders changes with every new release. Some contributors only submit a single patch that fixes a bug they discovered, others participate long term and may even evolve into core developers or maintainers of projects. As all Open Source licences guarantee the same minimum set of freedoms to all users, the question of who owns the code in an Open Source project may be more relevant for the individual reputation of the contributors than to the adoption or the value of the software. Many Open Source developers care strongly about being properly attributed for their work as this builds their reputation and their personal value, despite there being minimal restrictions on the distribution of the software they created.

The contributors who currently develop and maintain an Open Source project are typically a subset of all of the copyright holders. The contributions of their predecessors facilitate the continued development of the software and the prior participants' licence decision will impact release of new versions. The current group of contributors assumes *joint stewardship* over the technical development of the project and the management of the community (whether or not they are joint owners).

To become an active stakeholder in an Open Source project, newcomers must engage with the community that develops it. Since contributor fluctuation is quite normal in Open Source communities, a key aspect of community governance is to ensure continuity in the event of changing participants. In communities that have been active for a period of time, the currently active stakeholders steer the project on behalf of themselves and the previous contributors who are likely copyright holders on the code, but who no longer participate in the project. In some organisations, being an active contributor or qualifying for membership in the technical or administrative steering bodies is tied to a financial contribution or organisational membership fee. These fees are nominal in communities but may be quite substantial in foundations and may depend on the size or turnover of the contributing organisations. They cover administrative, governance, and community management costs. The software the community produces is still Open Source and free to use, irrespective of such fees. Governance and Open Source licensing are two separate dimensions.

5.7 Communities, contributors and merit

The term *community* is crucial to Open Source but can confuse as it is used with different connotations. To understand how Open Source is produced, it is necessary to distinguish Open Source communities from other organisations and to derive the specific functions performed by communities from this. Some people intuitively assume that Open Source communities consist primarily of enthusiastic volunteers collaborating for the common good. This is, however, not necessarily the case, and increasingly businesses and paid developers make up significant proportions of community.

The mix of different types of contributors like individual volunteers, businesses, or community staff is referred to as *community composition*. Quantitatively, most contributions originate from businesses and the majority of individual contributors participate in Open Source development as part of their employment,[123] making the communities *hybrid* or *heterogenous* in their composition. While licensing defines whether a piece of code is Open Source or not, the *openness* of a community is defined by its governance norms. Communities with many diverse participants typically prefer open, transparent, and accountable governance processes.

Producing Open Source is of course a necessary aspect of any Open Source community, but not sufficient for a definition. Different entities, state authorities, or anonymous donors may release code under Open Source licences without ever engaging with others to build a community.

Open Source communities differ in *how* their products are created, which is a question of governance. Especially volunteer-driven communities care strongly about the values they communicate to their potential participants. The KDE Manifesto, for example, lists open governance, free software, inclusivity, innovation, common ownership, and end-user focus as essentials.[3]

A *contributor* is an individual or organisation that invests resources in the creation of a community product. If a contributing developer is employed by an organisation, the employer pays the salary and may gain the copyright on the contributed code. In many organisations individual developers are allowed to contribute in their own name. This creates a lack of clarity as to whether an individual contributes on their own account, that of an employer, or one of a number of organisations with whom they are associated.

In the case of doubt, from a governance perspective the entity to be considered the contributor should be the one that has the authority to decide what effort to contribute and to what product or community. Depending on the internal arrangement, this may be the individual or the company.

There are many different types of possible *contributions*, not just the provision of code. Other examples are organising a community event, translating software to local languages, maintaining the community web

[3] https://manifesto.kde.org/ (accessed 03/02/2024)

site or writing newsletters. The different types can be normalised to contributions of time, money, or knowledge in the form of experience or expertise. By participating, contributors gain *merit* in the community which defines their standing within their peer group. Merit develops organically based, for example, on technical expertise, long-term commitment, or the social role and reputation of the individual. Merit or the value of contributions are at times difficult to measure in quantitative terms.

While Open Source licences govern the contributor–user relationship, governance structures govern the contributor–community relationship.

Contributors participate in Open Source development and engage with the community voluntarily and based on their own desires. Even though anybody can use, study, modify, and redistribute Open Source, nobody can be forced or coerced to contribute to it, and arguably there is no moral imperative to do so. Any individual or organisation that contributes decides that it is the right thing to do for them.[83] Engagement in Open Source communities may be explained based on Hirschman's concept of exit, voice, and loyalty. Hirschman researched consumer loyalty based on "a conceptual ultimatum that confronts consumers in the face of deteriorating quality of goods: either 'exit' or 'voice'".[65] Long-term consumers of a product develop loyalty to it in that they would rather continue to use it and do not wish to change to a different one. When applying Hirschman's concept to Open Source, the change of quality is that of the governance of the community, while the loyalty is that of a long-time participant. The option of having a voice in Open Source comes from engaging in the community to maintain its quality and to participate in joint stewardship. The exit option is to stop participating and to disengage from the community. Over time, contributors tend to feel very strongly about their communities and develop a sense of belonging. When considering whether or not to continue to participate, they often prefer not to let their fellow contributors down.

A specific form of exit in Open Source occurs where there is a fork. A fork is a split of the development community where a group of contributors establishes a new 'centre of development'[4] to continue the development of their own separate version of a product. Well-known forks are the LibreOffice/OpenOffice, the Elasticsearch/Opensearch and the Own-

[4] https://opensource.com/article/19/1/forking-good (accessed 03/02/2024)

Cloud/Nextcloud splits. Forks are a corrective measure that ensures that community governance stays aligned with the interests of the currently active contributors.

They are made possible by the essential provisions of Open Source licences and almost always represent an issue with community governance, illustrating further the duality of Open Source licensing and community governance as separate dimensions. Forks come at a cost, for example in the form of a split of the contributor base, added technical complexity, or interoperability issues, which is why contributors do not take this step lightly.

This fact focuses the community on whether there is a less destructive way of tackling the issue which instigated the possibility of forking in the first place. Accordingly, the threat of forking provides additional checks and balances over how the community governs itself.

Voluntary participation together with the *potential of forks* keep contributor interests and community governance in line.

The combination of licensing and governance provides a suitable definition of what makes an Open Source community: An *Open Source community* "is ... a social group of contributors that participate voluntarily in the production of public information goods"[21] The two functions that communities need to provide based on this definition are:

- *community governance*, which determines the perceived quality of the organisation in the eyes of the contributors and influences their voice-or-exit decisions; and

- *community management* as the task to motivate contributors to join, actively participate in the community, and to stay active instead of exiting.

Contributors join communities if they expect to achieve their goals more easily as part of the group compared to working alone. This comes as a trade-off, as to become a part of the community, a share of the contributor's investment needs to be directed towards being a community member as opposed to the development effort.

The term *governance* refers to all of the processes of social organisation and coordination within the group. Essential aspects of community governance include the explicit and implicit organisational structure, the decision-making and conflict resolution processes, and the social order of the community.

The reasons why communities experience contributor fluctuation can be conceptually separated into changes in motives and changes in motivation. The motives of a contributor may change based on external developments. People may graduate from university where they enjoyed Open Source development, or start a family and intend to spend more time with their kids, or change jobs and now work on different things. Communities need to accept and possibly even encourage such changes as a sign of a healthy personal development. Changes in motivation, however, are caused by internal community processes that affect organisational quality as perceived by the participant. They are determined by the community's governance norms and maintained through community management. How the community makes decisions and resolves conflicts, the impact of speaking up to influence the group, how decisions are enforced in the face of voluntary participation, the support the community provides to the creative development process, and the delineation of community members and outsiders all influence the perceived quality of the organisation. Contributors participate voluntarily in Open Source communities. They engage in the production as well as the governance processes. Communities facilitate contributions through their governance and attract contributors through community management.

5.8 Value at the edge of the commons

At the intersection of Open Source and business there is an apparent *tragedy of the commons*.[60] Developing Open Source is a virtuous effort that contributes positively to society by improving the available technology stock. At the individual level, contributors are passionate about their work and love what they do. On the other hand, many businesses that aim to create value by developing and building upon Open Source technologies struggle to find viable business models. Some have questioned the sustainability of the Open Source development model as a whole even though the wider Open Source community is thriving.

One source of confusion in this context is the much-repeated question of ’how to make money with Open Source’. This question is difficult to answer because it reduces benefiting from Open Source to capturing value through the generation of revenue.[25] It has long been established that there is no special Open Source economy.[154] Instead, the production of Open Source follows basic economic principles in a process that can be explained by breaking down more systematically how businesses benefit from Open Source.

5.9 The global upstream/downstream network

To illustrate how to position businesses in the Open Source value chain, it is necessary first to look at how the wider Open Source community organises itself. Individual communities develop specific Open Source products representing parts that need to be integrated into consumer-focused software and hardware products in order to be useful. The mechanism that coordinates the efforts of the various specialised communities is called the global upstream/downstream network. This network integrates the work of the various specialised communities into a technology stack suitable for end-users or as platforms for commercial products. This upstream/downstream model of collaboration within the wider Open Source community uses the mental image of a large river that collects the water from many tributaries (the communities) and delivers it to the ocean (the users).

No central decision-making body exists to coordinate within the global upstream/downstream network; instead, the communities operate autonomously and react to the stimulus from feedback and contributions in a competition for relevance and adoption of their solutions. Product improvements originate in the communities and are integrated ‘down the stream’ by more and more complex aggregated products. Feedback such as bug reports and requests for improvements, but also patches meant for integration into the upstream projects, are generated closer to the users, and then travel ‘up the stream’ to be eventually integrated by the originating community for that package. Practically all relevant Open Source solutions are part of this network that coordinates between supply and demand of Open Source contributions, resulting in complex, highly integrated products, for example Linux distributions or device platforms like Android or Yocto.

All copyright licensing within the wider Open Source community is automatic and transitive. It is automatic in that no negotiation takes place between the authors and the users of the software, and use of the software is subject to the terms on which is it is licensed. Licensing is transitive in that everybody in possession of the software is able to redistribute it to make it available to any other party without the need to refer back to the original author. By way of the combination of automatic and transitive licensing, the wider Open Source community avoids potential 'anti-commons' situations. In an anti-commons situation, the effort to acquire all necessary IP becomes prohibitively high, resulting in an underuse of available assets.[62] The widespread reuse and integration of Open Source solutions in the upstream/downstream network potentially results in an exponential increase in the number of licensing relationships that can easily produce an anti-commons situation. It is therefore essential for the functioning of the global upstream/downstream network that all necessary rights are acquired *ex ante* and without the need for negotiation. This is one reason why patent holders have found it difficult to combine the use of Open Source with revenue-bearing patent licensing. To avoid the necessary negotiations with patent holders after the software has already been used, the Open Source community tends not to adopt patent encumbered technologies.[15]

5.10 Open source-related products and services

Businesses operating in the Open Source ecosystem offer a combination of *goods* and *services*. These terms are loaded with different meanings, for example based on whether what is sold by the business is a unit of a good or billable hours. To avoid confusion, the distinction made in our context is that offering a product requires the right to do so, for example based on the ownership of physical goods or the rights to sell commercial software licences or redistribute code, while services can be sold in a way that they complement a good somebody else possesses.

Applied to Open Source, businesses must make a choice regarding how to manage IP related to the value proposition they offer to consumers. To pursue product-based business strategies, they must retain appropriate rights over the complete product source code, for example through the application of CLAs.[89]. These agreements ensure that the business has the necessary rights to sell proprietary licences to the software, at the expense of creating asymmetry between the contributors to the software: While those participants that submit improvements and bug fixes under the CLA can contribute to and use the software as Open Source, only one entity has the rights to benefit commercially from it. Such 'single-vendor' business models require a strong market position usually based on thought leadership and innovativeness that convinces external contributors to participate.[122] MySQL, Qt, or Asterisk are examples of products that have been successfully developed under such a goods model. In contrast, offering services related to Open Source products does not require appropriation of the software by the vendor of the service. Anybody with the necessary expertise may offer support, custom development, or operate Open Source-based solutions for clients without the need to own the copyright to the original code and companies may offer support for code that is distributed by others.

Using this distinction between goods and services as the basis for revenue generation, Open Source-based value propositions can be further broken down based on their function in the value chain. Software products may be used as foundational technology or as consumer technology.

Foundational technologies are the intermediate products or building blocks of the tech sector that provide common functionality from the operating system to web-based communication or user interface frameworks. They are combined by manufacturers into more consumer-ready devices, even though the consumers often do not know or necessarily care which exact software the device contains so long as it does the job.

Consumer products are made to satisfy concrete needs rather than for reuse or as means of production. Open Source solutions such as operating systems, a boot loader to prepare the device and start it running, programing language runtimes or databases are common building blocks used in many devices that consumers expect but are usually indifferent to. Even most proprietary software regularly also includes common Open Source modules for these.

Businesses may offer either *vertical integration* where they operate

Open Source as a service for their customers or offer services that are *complementary* to the product itself.

Building custom websites based on an Open Source web framework vertically integrates the framework into a higher-level application. Similarly, a cloud provider that provides instances of an Open Source database vertically integrates the software into their main product, the operation of data centres. To provide vertical integration, the service vendor requires expertise on how to use the underlying software modules and domain knowledge about the intended application, but – unless the vendor wishes to extend their functionality – not necessarily knowledge on how to develop the software. Vertical integration creates value in the eye of the consumer on top of the underlying Open Source solutions.

Complementary services in comparison focus on supporting the Open Source modules themselves, as in custom feature development or long-term maintenance of a library. Instead of building higher-level functionality, they make foundational technologies viable for inclusion into other products or for use in different domains. Since the Open Source ecosystem is built upon the idea that everybody can maintain and extend the software, complementary services are an essential element of it. Many Open Source developers make a living from being the main contributors to software modules and getting hired to improve or extend them. Such services are not free. The philosophy that the software should be free while it is at least partially developed commercially is an inherent idea of Open Source. Even selling copies of the software is explicitly allowed by the terms of Open Source licences, which is represented by the slogan that 'Open Source is commercial'.[5] Instead, the Open Source community distinguishes between Open Source and proprietary software based on the licences applied.

Vertical integration requires a higher grade of domain knowledge, while horizontal complementary services require more technical expertise about how a specific software module is implemented. The vendors involved similarly develop into different roles. Vertical integrators often regard the Open Source modules used as stable and complete and focus on providing them to a wide range of customers. Horizontal service vendors care about the software as something to maintain and improve.

[5] https://www.gnu.org/philosophy/selling.html (accessed 03/02/2024)

They contribute more changes to the software compared to the vertical integrators, sometimes leading to criticism that those only use the software without 'giving back to the community'. In particular, businesses that pursue growth strategies based on Open Source technologies, sometimes funded by venture capital, find it difficult to convert vertical integrators into paying customers. Since there is no obligation to contribute back or to 'pay a fair share', these difficulties illustrate problems in the underlying business models rather than with the sustainability of Open Source development. To be able to offer convincing value propositions to potential customers, businesses need to maintain a level of control over the goods they are selling. Open Source in itself as a public good does not provide this leverage.

Businesses have tried to combine Open Source with various approaches to capture its value. The single-vendor model mentioned previously is one of them. Others include *trademark* licensing programs, the acquisition of *patents* parallel to software development, or attempts to control the market by enforcing the use of standards covered by SEPs.[91] All of these approaches represent a trade-off between software freedom and capturing value in a business.

In summary, the Open Source-based value propositions can be broken down into:

- foundational versus consumer-oriented products; and
- vertically integrated versus horizontally complementary services.

The vast majority of well-known businesses in Open Source sell services, a fact sometimes obscured by services being marketed in a way that is similar to physical goods.

The subscriptions to technical support that Red Hat offers are a service complementary to the Linux distribution that contains thousands of software package developed by the wider Open Source community which Red Hat does not need to own. GitLab (the company) offers hosted software development infrastructure as a vertically integrated or support for on-premise installations as a complementary service based on GitLab (the software). GitHub hosts many important Open Source projects, even though it offers a vertically integrated service that itself is not free software. Android phones are proprietary consumer products based on an

Open Source foundation, the Android Open Source Project.[6]

5.11 The benefits of Open Source in a business context

The success of Open Source has incentivised entrepreneurs to build revenue streams based on it.[25] However, 'making money with Open Source' is only one way a business can benefit from Open Source. With regards to business strategy, there are three possible scenarios regarding how Open Source can be useful to businesses:

- Open Source can be useful without the goal of direct financial benefit, or
- it can be used to directly to *generate* revenue by being sold, or
- it may provide a way to *reduce* the cost of a product.

All three scenarios are relevant in practice. Many software engineering tools like compilers, build systems or programing language environments today are developed in an Open Source-first approach. These tools enable ecosystems of specialised functionality and drive developer engagement through knowledge transfer. By standardising their ICT infrastructure and engineering toolchains, through Open Source, a business can significantly reduce up-front expenditures and focus the R&D budget on consumer-relevant product functionality which it believes drives consumer choice. Similarly, Open Source enables interoperability, for example through shared application programming interfaces (APIs), which allows independent organisations to build solutions that integrate with each other.

A different approach is the loss-leader – software products like web browsers that are distributed as Open Source to attract consumers and market other value propositions. All these approaches benefit businesses indirectly without generating revenue for them.

To build revenue streams, businesses can offer products or goods for licensing or use them to offer integrated or complementary services.

[6] `https://source.android.com` (accessed 03/02/2024)

In the single-vendor case, all rights to the source code are controlled by the vendor, enabling them to license the same source code to different customers under different terms. This dual-licensing or multi-licensing approach represents a price differentiation mechanism that allows vendors to achieve a larger market share through the additional adoption of those consumers who choose the Open Source solution out of a preference for the licence or to save cost. The product is typically offered under a copyleft Open Source licence that requires customers to disclose their own source code, with the alternative offer to buy commercial licences to avoid this requirement so promoting the uptake of commercial licences to avoid concerns, with respect to the use and impact of such copyleft code.

Because the necessary CLAs create asymmetry between the contributors, this approach is controversial in Open Source communities, as was seen in January 2021, when Elastic moved two of its products from Apache 2.0 to the proprietary SSPL licence and was able to do so only because it had received CLAs from its community of contributors.

For a company to achieve a reputation in the market that enables it to apply multi-licensing approaches requires thought leadership and innovation to build the necessary goodwill with both contributors and consumers. Few are successful and failing companies risk to losing external contributions and may end up carrying the development cost of the complete product. There is also a risk of forks, which was the consequence in the Elastic situation.

A third benefit is cost reduction in building software or devices. In the simplest case, substituting a proprietary software module with an equivalent Open Source implementation eliminates the licensing effort and importantly cost or royalty.

More commonly, the required functionality is needed by many companies but not readily available in the market. This incentivises businesses to pool R&D cost with others that have the same needs, effectively reducing their own investment to a fraction of the overall cost. In such a setup, participants generally expect to be equals amongst the other contributors. Open Source licensing facilitates the collaboration. Projects are commonly set up as not-for-profit organisations or at Open Source foundations and typically industry-driven.[128]

The same company may develop different parts of its product portfolio under different models. It is quite common that manufacturers compete in the same market segment with their products while at the same time collaborating in Open Source projects. Since the Open Source product itself is always free to all interested users and for all purposes, a "good business model is simply one that succeeds in creating additional value at the edge of the commons".[154] One essential question for every Open Source-related strategy is how the business benefits from its participation in the Open Source ecosystem by one of these three approaches:

- by generating revenue; or
- by reducing cost, or
- by realising other non-financial benefits.

5.12 Differentiating in the eyes of the consumer

Businesses decide where to compete and where to collaborate based on what they expect to be differentiating product features that convince consumers to prefer their products over those of their competitors. Common product features are best implemented using existing Open Source solutions both to share R&D cost with other contributors and to benefit from the joint expertise of the stakeholders involved. The differentiating product features are more commonly developed inhouse by the vendors themselves and not Open Source. Embedded or mobile devices today share a large part of their foundational software modules, while they implement user experience and application-specific business logic elements as proprietary software.

This differentiation is exclusively in the eyes of the consumer. There are two common logical fallacies:

- First, contributors assume that because they invested time and resources into developing a product, they are entitled to compensation. Unfortunately, it is quite common that businesses make the wrong bet and develop products that are not convincing to consumers. In an open economy, the 'fair compensation' a business should expect as the return on R&D investments is the value the market assigns to the product. Only by focusing investments towards those product features that consumers value can a business be successful in the market.

- Secondly, Open Source developers sometimes expect that because *they* have contributed valuable code, consumers should work with *them* and hire them for example for ongoing development or operational support. There is, however, no intrinsic value for the consumer or a vertical integrator in a business relationship with the core developers of a software unless this relationship benefits both sides beyond the free licence to the software, for example by adding value through knowledge transfer.

Developers and businesses that produce Open Source solutions must find ways to differentiate their value propositions. Positive differentiators include:

- the perception of quality and innovativeness based on the joint expertise of different stakeholders who participate in its development;
- enhanced trust based on the ability to verify the functionality and integrity of the software;
- the sustainability of the development model;
- reduced lock-in; and
- the option to procure maintenance and feature development work from a variety of providers and other factors that promise that the software is useful to the consumer in the long term.

Many of these positive connotations that Open Source vendors can utilise to differentiate are influenced by the impression of community health, which is commonly assessed with metrics like the number of independent organisations and individuals participating in the Open Source development process or the overall number of contributions raised by the community. These metrics reflect negatively on single-vendor models and partially explain the hesitation especially of vertical integrators to engage in a business relationship.

While the assumption that corporate Open Source users in particular should return a 'fair share' to its developers is understandable, there is no imperative to contribute from the individualistic economic perspective (it could be explained by other disciplines).

However, conflicts based on free-riding behaviour are rare in the Open Source ecosystem, and almost all involve Open Source vendors that implement the single-vendor model.

Some Open Source vendors, particularly those with venture capital (VC) funding criticise their users and the community for not giving enough back to them. This 'community bashing' is reminiscent of politicians that criticise their electorate for exercising their free will not to support their policies. More rationally, the behaviour of the consumers can be explained by the negative differentiation effects that is caused by the reintroduced vendor lock-in or the lack of sustainability of the software development process caused by the absence of a healthy, diverse developer community.

In short, some single-vendor Open Source businesses attempt to reintroduce proprietary software development models and strong vendor lock-in based on an Open Source product, which reduces software freedom. These attempts contribute to the negative reputation of contributor licence agreements that the vendors require to pursue these strategies.

5.13 The role of the volunteer community

In the discussion of the economics of Open Source, the focus is less on the volunteer community since it contributes only a fraction of the overall Open Source development effort and is less engaged in the larger Open Source foundations and the discussion of commercial models. The volunteer community is, however, a core element of the wider Open Source ecosystem. This is illustrated by the early success and market adoption of Open Source that happened when business participation in community development was still considered an extravagance. The volunteer community acts as a driver that stimulates Open Source innovativeness.

The individual motivation of volunteers to participate in Open Source development differs from the business rationale outlined earlier. Developers start contributing to Open Source projects based on their own interests and technical needs and over time evolve a deeper engagement with the developer community to gain a sense of achievement and belonging. Open Source culture builds on the careful, transparent, and consensus-focused governance that the communities set up based on the paradigm of voluntary participation and the correctives affected by the possibility of forks. Many contributors consider Open Source development something worth fighting for. They focus more heavily on the virtuosity of contributing and the benefits of Open Source to society. Software freedom has more importance to them than the availability of the source code. As such, the engagement of the decentralised Open Source developer community provides an important safeguard of software freedom and represents the interests of civil society.[21]

From the perspective of participating individuals, the positive freedom to use, study, modify, and redistribute the software gains more emphasis compared to the absence of constraints. This perspective has defined the debate on openness and freedom indicative in the histories of the FSF and the Open Source Initiative as opposed to the industry-led Open Source foundations. Representation of the decentralised volunteer community of Open Source contributors often focuses on charitable goals and is separate from industry-led foundations which are essentially trade associations. This makes volunteer community organisations natural counter-parts for policy-makers and gives them a sometimes oversized credibility and reputation as trusted advisers. Compared to the industry associations, volunteer organisations operate on small budgets and staff. There is a generally fluid transition from being a volunteer contributor to a corporate one upon graduation or with the creation of start-up businesses, emphasising the importance of Open Source as a knowledge transfer mechanism. The volunteer community is an essential and necessary part of the wider Open Source ecosystem that supports the competitiveness of the software market and the alignment of technical innovation with the interests of society.

5.14 Competition in the wider Open Source community

As there is generally no direct remuneration for Open Source contributions, competition in the Open Source ecosystem is not about revenue or market share. Contributions are not market transactions in which two parties negotiate a trade they assume to be of similar value for both sides. However, the Open Source community exhibits a fast pace of innovation, develops new state-of-the-art technology, and swiftly reacts to changes in the technology needs of the consumers. There is competition within the communities, between the communities in the Open Source ecosystem and with the rest of the market and even government.

Within the communities, contributors compete for the integration of their code to be released with the community products. Since the motivation of individual contributors is often driven by non-monetary factors like a sense of achievement, positive creativity, or pride, the effort that is invested into an incremental improvement of a specific feature is at times higher than justified purely by technical requirements. This perfectionist attitude of 'it is done when it is done' enhances overall product quality. The prestige of proven contributions to important Open Source projects or a good reputation as a contributor is valuable enough that in the software sector, they come to be considered a part of the developer CV. This combined with the potential global participation of individual developers makes for a rather fierce intra-community competition.[162]

Open Source communities compete with each other for the adoption and integration of their solutions within the global upstream/downstream network and eventually the consumer market. This mindset drives the acquisition of new contributors and the continued development of the software that defines the relevance of the community and its ability to facilitate contributions and raise funding. Participants in Open Source communities bet on the adoption of their community's software to help them realise the benefits from their contributions. This inter-community competition for relevance and adoption causes swift technological cycles that displace even well-known solutions with a large contributor base once a more promising alternative emerges, as illustrated by the competition between the OpenStack and Kubernetes communities. The participants in these communities frequently stay involved and continue to contribute to Open Source, however they quickly shift attention and contributions to the newly dominant solutions. While individual communities grow and shrink, the overall community of Open Source participants seems to mostly grow slowly and steadily. The upstream/downstream network exhibits powerful positive externalities of community health and size and fast adoption of new technologies similar to the tipping markets of Internet products.[133]

When Open Source products substitute proprietary products, the community competes with private enterprise. The inherent freedoms and public good character of Open Source make it difficult for businesses to compete since they need to offer strong value propositions to justify a non-zero price. This kind of one-to-one competition between free and commercial products was however more common in the earlier days of Open Source adoption as a way to challenge the market position of entrenched incumbent software vendors, resulting, for example, in LibreOffice, the Linux kernel, and Apache. Collaborating on Open Source solutions challenged the market position of the incumbents and forced their products to be more consumer-oriented, without necessarily replacing them. Markets for some proprietary products have practically disappeared, such as those for commercial software development tools or proprietary embedded operating systems. In these cases, collaboratively developed products perform better based on a wide stakeholder participation in the development process. Today, especially industry participants have adopted a conceptual separation between a competitive zone where consumer-oriented, differentiating product features are developed and a collaborative zone that creates the underlying non-differentiating functionality. Different behaviour is expected in these zones. The competitive zone covers a smaller part of the overall software of a device or application and functions mostly unchanged in terms of development processes and IPR management compared to traditional R&D. In the collaborative zone, a key principle is joint stewardship over the community products with the expectation of ex ante licensing of all relevant IP. Based on this collaborative approach, businesses engage in a model of continuous non-differentiating cooperation.[15]

An under-researched aspect of Open Source community collaboration is the way it competes with state actors. Open Source offers technical solutions that may challenge established political processes, as for example in e-democracy applications, and facilitates new forms of political participation. The production of private goods is coordinated primarily within firms or between firms in markets, while public goods are mostly provisioned by the state.[32] Open Source enables a 'fourth transactional framework'[11] that provisions public goods in an alternative, decentralised fashion.

The cross-border collaboration of Open Source communities also challenges established policy frameworks, which is of particular relevance with Brexit and other geo-political shifts in the US, China, and Europe, in particular calls for digital sovereignty. However, local legislation like the European General Data Protection Regulation (GDPR) and governmental policies such as the Chinese 'Great Firewall' still have significant impact. There is not much research on the potential competition between the Open Source community and government at this juncture. This may, however, become an important field of inquiry for both developing and developed countries, especially as the impact of Open Source correlates with policy objectives like the United Nations Sustainable Development Goals.[7]

Overall, the wider Open Source community represents a very competitive environment that results in fast-paced technical innovation, reduced barriers to entry, and a challenge to incumbent market positions. The introduction of Open Source both from a licensing and a collaboration perspective usually increases competition by providing alternative models and approaches. Open Source community governance norms are, however, not fully standardised, with a theoretical possibility that participants may form market-controlling clubs and it is normal to have antitrust or competition policies to avoid this. The public good character of Open Source combined with open, transparent governance norms generally inhibit possible anticompetitive behaviour. A key contribution of Open Source to economic growth is the provision of baseline technologies that represent the current state of the art and are available to everyone.

[7] https://sustainabledevelopment.un.org/ (accessed 03/02/2024)

5.15 Compliance, social and market transactions and zero price

Key effects of Open Source collaboration include:

- the reduction of transaction cost of participation and
- the reduction of barriers to entry for newcomers.

Reduced transaction cost opens the collaboration process to an overall larger constituency and in particular invites participants for whom staff cost and membership fees, for example in traditional standards development have been a challenge, namely SMEs.[15] Reduced barriers to entry for newcomers improve the chances for example of university spin-offs and start-up companies to compete with incumbents, improving competitiveness.[98]

These benefits depend on efficient IP management across the whole Open Source ecosystem. An implicit expectation of *Open Source compliance* towards all participants requires adherence to the obligations from all consumed Open Source licences and is regarded as a *hygiene factor* by the wider Open Source community. Hygiene factors create dissatisfaction by their absence.[64] In the context of Open Source, compliance is considered as a tool of the trade. Uncertainty about the compliance of suppliers and consumers or necessary litigation undermine the fabric of the global upstream/downstream network by negating the reductions in transaction cost of participation and reerecting barriers to entry.

Open Source is made freely available to everybody. A price of zero triggers a different response in actors even compared to a bargain low price. It transforms the exchange between communities and consumers from a market transaction where parties negotiate for gains at the others expense to a social exchange with an agreement on behaviour that is beneficial to everybody involved. Asking for remuneration or negotiating over commitments constitutes anti-social behaviour in a social exchange. Instead, there is an expectation of fairness and reciprocity.[4] This explains why Open Source compliance should not be managed as a risk but considered an imperative.

Reciprocal licences model Open Source usage as a social transaction by asking the consumer to act similarly to the licensor. They aim at symmetry between contributors and users and at ensuring that using the software remains a social exchange for the long term. Because it is intuitively understood that even a small number of bad actors reduce the overall willingness to engage in social exchanges, the Open Source community is sensitive to licence violations and willing to litigate against them.[8] This attitude of the wider Open Source community is in line with recent research that shows a weakening of social norms in the face of even a few bad examples.[4] Considering Open Source participation as a social exchange again helps to explain the negative reputation of CLAs since they are based on market exchanges. Contributing as an individual to a single-vendor product under CLA is more similar to paying taxes than to bringing a dish to a dinner with friends.[11] Similarly, programs that put out bounties for adding features or fixing bugs reframe contributing to Open Source as a market transaction.

A framework for social transactions is disrupted as soon as any party starts negotiating about remuneration, which is why the Open Source community avoids any form of negotiation of terms between contributor and consumer. Instead, Open Source licences are *ex ante* agreements where all rights necessary to build the desired outcome are secured from the start. The agreements are standardised into a number of approved licences that model the three basic modes of Open Source collaboration:

- strong copyleft,
- weak copyleft, and
- and permissive.

Only a small number of licences are used in recent practice and new licences are rarely approved. Combining Open Source licensing with other rights such as patents or trademarks that require *ex post* licensing of terms has generally not been successful.[15]

Maintaining licence compliance has gained considerable complexity due in particular to modern hardware devices such as general-purpose computers that include all sorts of computing, storage, user interface, and networking functionality.[63] Manufacturers are required to document the complete use of Open Source in the devices according to the

[8] `https://gpl-violations.org/` (accessed 03/02/2024)

obligations from the licences contained in them. Recent initiatives aim at reducing this complexity to ensure the viability of the Open Source supply chain.[9] Industry and community best practices in this context continue to evolve.[10]

5.16 Open Source as community-provisioned public goods

Open Source affects society at all layers of the economy. Individuals participate for their own reasons and learn valuable skills. They find enjoyment, a sense of achievement, and belonging and opportunities to be a productive part of something bigger: the wider Open Source community.

Businesses find both threats and opportunities from Open Source. It greases competition and fosters the innovativeness of the tech sector by providing free baseline technologies, potentially weakening market positions of incumbents and causing creative destruction of existing assets. On the other hand, Open Source offers plenty of business opportunities and faster, cheaper ways to produce modern applications and devices.

For every Open Source-derived product a business intends to market, it needs to answer (at least) these three questions to describe the value proposition to the consumer:

- What is the revenue model of the product? Is the goal to provide a defined functionality at the lowest possible cost, or to induce indirect benefits without directly generating revenue, or is the product meant to be sold in the market to establish a revenue stream?
- What type of Open Source-related good is it? Is it a service of either the horizontal complementary or the vertically integrated kind, or is it a product which may either consumer-oriented or a foundational technology?
- What is different about what their offer? The business needs to market the product to the consumer based on differentiating product

[9] `https://www.openchainproject.org/` (accessed 03/02/2024)
[10] `https://reuse.software/` (accessed 03/02/2024)

features, while including the expected but non-differentiating features at low cost in order to sell the product at competitive prices.

Based on the answers, participants can derive ways to build and market their products. Some combinations represent well-known approaches. Kubernetes and Linux are non-differentiating foundational technologies that reduce the cost to operate data centres or build devices. They are developed at Open Source foundations in a model where participants pool R&D cost. ChromeOS and clang are differentiating consumer technology that realise indirect benefits by enabling developer ecosystems or marketing complementary services. They are developed as Open Source by a single company or a small group of stakeholders. Android phones and proprietary apps in general are differentiating consumer-oriented products intended to generate revenue. Their differentiating features are developed in-house while they build on foundational technologies developed in collaboration with others.

Not all combinations of answers have been successful in the market. Building differentiating consumer-oriented products as Open Source that are supposed to directly generate revenue lacks the necessary differentiation in the eyes of the consumer. These approaches suffer from an inherent contradiction that the purpose of a business is differentiation in the market, while the essence of Open Source is to be non-differentiating. An alternative is the Red Hat model to differentiate based on services that complement the free product. By answering how they create additional value on the edge of the commons, businesses can embrace Open Source and benefit from it.

From a macro-economic perspective, Open Source is a toolbox that is part of the available technology. Software only has a tangible effect if it is executed in an application or device. Because of that, there is no possible economic downside from the development effort itself of the wider Open Source community at the macro level, while Open Source is proven to be useful and drive innovation. Hence Open Source *contributes positively* to the common good.

Where Open Source participation is regarded as a social transaction, free riding is not common. By resolving the free rider problem, Open Source development processes open up the possibility for the decentralised collaborative development of public information goods at a large scale.

Open Source benefits society by combining innovative state-of-the-art technologies that are at the same time commodities and public goods and can be created without the need for a central authority. As such, Open Source has become a part of the political sphere. Political systems should be designed so that they serve society even if the stakeholders involved— citizens, politicians, businesses, and others— act in their own self-interest. Open Source provides the tools and processes to make this possible for the creation of software technology and information goods in general. It fills an economic gap by enabling the decentralised provision of non-market goods potentially at a global scale. By connecting individuals and organisations in the production process, Open Source bridges the formal and informal economy. This enables grassroots and volunteer initiatives to have an impact, as illustrated by the successes of the early volunteer-driven communities. The mechanisms described in this chapter – types of goods, competition versus collaboration, differentiation – are agnostic to economic systems and observable in all of today's societies. There is however a direct relationship between software freedom and individual freedom – self-identification, the way Open Source contributors choose what to contribute and where— depends on civil liberties.[118]

FOSS provisions public goods in the absence of centralised authority. There is a long way to go to realise this ideal, but the first steps are complete. A good next step will be to stop asking 'how can I make money with Open Source' and start asking 'How can I, my business or society realise the benefits from Open Source?'

Chapter 6

Conclusions

6.1 Contributions

If there is one overarching theme to the prior chapters, it is that as society puts more emphasis on knowledge and creativity, FOSS as a method of creating and sharing knowledge has almost universal impact. Aspects of governance influence access to the means of FOSS production. IPR regimes shape how knowledge is transferred and how one person can build upon the work of another. Standards development influences how the benefits of scale and the efficiency gains of agreeing to and widely adopting technology should be realized. This chapter provides a condensed overview of the insights contributed by the individual papers.

6.1.1 On open source community governance

The governance of communities has evolved as FOSS became more and more widely adopted. However, the mindset behind community governance as well as the social norms of the wider open source community are still strongly influenced by the spirit expressed for example in the Debian Social Contract and in software freedom. These starting points enabled collaboration focusing on individual needs first ("scratch your own itch"), followed by the subsequent integration of the resulting many small improvements into a coherent whole in the form of the numerous components that make up the upstream/downstream network. Engaging in a community of makers (see 2.3.1) remains one of the strongest motivators for individuals as well as employees of contributing organisations to engage in FOSS communities. Being able to work with and learn from like-minded peers creates a sense of belonging that transforms into a deeper connection of loyalty over time (see 2.3.4). Contributors "come for the technology and stay for the people". Self-identification by contributors with the tasks they wish to contribute to helps to satisfy the motivational need for positive challenges, even though it makes it more difficult to sustain maintenance work and bug-fixing.

Equality of opportunity among the peer group of contributors (see 2.3.2) ranks highly in the expectations of participants. The greater emphasis of diversity, equity and inclusion in the creation of equity and agency in communities today can be traced back to it.[27] Governance norms that emerged in early-stage volunteer driven FOSS communities continue to influence the community management of today. While contributors initially engage with communities who are the stewards of technologies relevant for their technical needs, the match of contribution opportunities and ethical convictions partially determines their longer-term engagement. Positive examples of outstandingly productive and welcoming communities also establish the governance standards that other projects are then measured against. The mantra of "who does the work decides" is present in today's expectation that technical project steering is separate from the membership-based project governance. Technical steering is meritocratic in the sense that technical steering committee (TSC) members are typically selected based on their track record of technical contributions. Equality among peers translates to an expectation that the outcomes are distributed as FOSS. For the products of a community, distributing the results as FOSS and in consequence for the community and participants to be FOSS license compliant is a hygiene factor, not a negotiable one. This insight helps to explain, for example, the negative reception of corporate attempts of ex-post license changes to a proprietary "business-source license" and the inclination to create forks of the affected products.[53] Similarly, an absence of discrimination and more generally positive diversity, equity and inclusion are also considered hygiene factors and expected as a given.[27] It is difficult to combine the spirit of open collaboration with a selective approach regarding who may participate. The wider open source community exhibits a strong preference for contributor symmetry, which also helps explain the reluctant acceptance of CLAs. Membership in a community facilitates product contributions, it is rarely a goal in itself. Participants join as contributors if they feel that more can be achieved when collaborating in a group than by working alone. However, community membership needs effort to maintain standing and reputation in the meritocracy, which in turn requires to divert a share of time to community engagement. For some, being a member of a community becomes a goal in itself. The resulting specialisation into makers focusing on product contributions versus community builders focusing on community management institutionalizes more formal organisational structures (see 2.3.3). This is even more pronounced today in cases where industry-led communities hire dedicated staff for example for program management or marketing from the start. Care needs to be taken that community management stays aligned with the productive activities and continues to serve the community. Over time, there is a risk of a disconnect between makers and community builders. Some examples exist of well-established institutions in the FOSS ecosystem which have historical relevance, even though it is not clear today what constituency they represent and how they gained and maintain or renew their mandate. This situation makes it more difficult to create consensus when representing the wider open source community, for example, to policy makers.[157] The ecosystem has at times resorted to a collaboration of the subset of open source foundations that as umbrella organizations host a specific set of projects or foundations. For these "code hosting organisations", it is transparent which communities and projects they represent. Similarly, in EU law, the concept of *open-source software steward* is now used to refer to organisations that are not manufacturers but ensure the viability of FOSS products.

Implementing truly open collaboration that is inclusive to all interested stakeholders is at the heart of most openly governed FOSS communities. However, barriers to entry and friction increase with the size and maturity of the community. Maintaining a welcoming, inviting culture as represented by KDE's "open doors policy" directly influences the perceived quality of the community and thus the inclination of participants to join, remain engaged and not to leave. A welcoming community governance is however not sufficient to keep contributors engaged. They also judge the usefulness and productivity of communities based on how ambitious the community's goals are compared to the state of the art in the wider open source community. In an idealistic sense, contributors want to make the world a better place and their contribution to make a difference in that (see 2.3.4). Not every contribution can be forward-looking and visionary, which is where the hybrid composition of most communities today helps. Corporate contributors are able to focus on a long-term road map, maintenance and security, while individual volunteers sometimes drive break-through improvements. The resulting balance between maintenance and new developments benefits the sustainability of the open source development model.

In collaboration that spans the globe, ethical principles and convictions will not be uniform. The wider open source community for the most part understands itself as an apolitical environment that functions based on partial consensus. It is sufficient if there is consensus between the contributors on the governance norms of the specific community and on the merits of contributions to the intended technical progress. Complete alignment on political, economic and societal goals of participation is not necessary. Achieving it would possibly be in conflict with the goals of openness and inclusion. However, even this understanding is not universally accepted, as parts of the community consider free software a social movement with political aims (see 5.1).

The basic needs for a community to function are inclusion, low barriers to entry, a positive communication culture and and commonly accepted personal conduct. At the same time, there is occasional resistance against formalization of behavioural norms for example by adopting a code of conduct. These difficulties in consensus building are reinforced by informal decision making. Meritocracy is regarded as a core tenet of FOSS culture and an overall positive influence. The term is understood so that the prestige and influence of contributors within the community should be measured by the aggregated value of their contributions, and nothing else. This is not in line or connected with the original use of the word.[10] In the context of FOSS, the wide application of meritocracy is seen as positive and successful. Despite the positive connotations, this at times naïve understanding of meritocracy disregards criticism of the term in other contexts that would also be applicable to the open source ecosystem. Its elitist undertones occasionally reflect on systematic distortions like the the male-dominated software development culture, the corresponding dominance of male contributors, the regional and north-south divides and the ableism of the equality of opportunity arguments. However, even considering the criticism, the wider open source community is still largely transparent, open and inclusive, and as such may serve as a role model for global collaboration.

As FOSS evolves from community-driven to industry-driven to potentially a regulated industry, FOSS governance has solidified and became increasingly normalized. However, even widely accepted norms have not yet been universally adopted as different communities and foundations continue to apply the practises that made them successful in the earlier stages.[142] With the trends towards hybrid community compositions and a rising share of corporate contributions continuing, the needs for formal organisational structures and explicit governance norms became more pronounced. In decision making and conflict resolution, community management still has to maintain the balance between voluntary participation and effective leadership. Instead of enforcing unpopular decisions, community management is an exercise of alliance and consensus building. Volunteer driven communities at times struggle to adapt their governance norms to this environment as they grow, resulting in difficulties to maintain growth in the later stage of community development.

The current environment of increasingly streamlined governance results in fewer and more competitive opportunities for contributors to become community leaders. Since the leaders of earlier stage communities are often the visionaries who build up the community in the first place, fewer contributors today rise to fame. Barriers to entry into established communities grow, both for technical contributions as well as for participation in community management. This may be one reason for the persistent observation that in many communities, few participants grow from consumption to be active contributors. There are two key reasons why FOSS communities need good governance (see 2.2). From a product perspective, it is required to coordinate the work of a diverse group of volunteers to create the community product. From a process perspective, it is necessary to maintain and grow the contributor base that forms the community. The importance of community governance cannot be overstated. There will always be new technical challenges to solve, especially with the scope of open source collaboration expanding into adjacent fields like data, hardware or artificial intelligence. However, governance determines the growth capabilities of FOSS communities and with that defines how complex, multi-faceted and simply how large the problems are that the communities can successfully solve.

6.1.2 On the relationship between FOSS and SDO

Beyond legal compatibility between FOSS and SDO IPR regimes lies the question of what positive motivation will lead the two groups to cooperate. To form a successful cooperation, both need to see each other as partners that can engage in a mutually beneficial relationship. The assumption that this relationship is a natural fit and always a positive one does not hold. Our research shows that in some situations, participation in SDOs or in FOSS communities should be considered strategic alternatives with regards to effectively facilitating standards, especially when participants have to choose an environment for collaboration and consensus building. Because of network effects, they prefer to choose one or the other, but not both.

In two steps, our research connects the worlds of standards development and of the wider open source community into a single, coherent theoretical model. First, the process of establishing a technical standard in the market was broken down into a model of four individual phases that both the standards and FOSS development exhibit (see 4.3.4). Second, two SWOT analyses have been performed where the strengths and weaknesses of SDO and FOSS were assessed separately against the common opportunities and threats in the ICT sector that both are exposed to (see 4.4.2). Anchoring the separate analyses in the same context with identical opportunities and threats allows to relate the individual strengths and weaknesses to each other.

The technical standardization process is triggered by a need for standardization. This need may be market based, for example to ensure interoperability between products of different vendors or to realize economies of scale in production. It could also be caused by a regulatory push, for example if new minimum safety standards are imposed by law. The process ends with the successful adoption of the new technology in the market in a way that satisfies the initial need for standardization. The standardization phase model maps the journey from the need for standardization to the end state of market adoption. We identify the separate phases of ideation, specification, implementation and diffusion of technical standards. While these phases can be observed in both standards and FOSS development, they may occur in different order (see 4.3.4). The models of specification-first, implementation-first, and parallel standardisation are defined as different sequences of the phases (see 3.5). FOSS collaboration often applies implementation-first or parallel standardization. Industry standards are explained as implementation-first technologies with informal specifications. Iterative approaches to standards development are explained as the repeated execution of the specification, implementation and diffusion phases. Our phase model is able to describe different approaches to standardization, including SDO processes, FOSS collaboration and industry standards setting as well as iterative approaches. In contrast, previously favored theoretical approaches that separate standards development and the subsequent introduction of standards-conforming products have severe limitations. First, these approaches require specification to occur before implementation, which is not empirically supported. Second, they also insist that standards are documents that are established by consensus and approved by a recognized body, even though such a technocratic definition clearly reduces the analysis to only a subset of all standardization processes (see 4.3.1). Our phase mode of standardization focuses on how markets adopt technical standards triggered by identified needs for standardization. It represents a utilitarian approach to standards development and tries to overcome theories that are based on historical path decisions like the establishment of national standards bodies and ISO. Overall, the theoretical foundation provided by this phase model is able to describe a much broader variety of standardization processes as they are observable in markets, from beginning to end.

Building on this common model for standardization processes, the strengths and weaknesses of SDO and FOSS communities help in understanding which approach may be better suited in what phase (see 4.4.2). Digitalization, the development of improved methods of collaboration, a general trend towards increased openness and transparency, the change in the role of the state towards being a hands-off regulator instead of an employer, and the shift from national to supra-national regulation have been identified as threats and opportunities that all innovation actors face in the ICT sector (see 4.5.1). In this challenging environment, standards and FOSS development exhibit differing strengths and weaknesses. FOSS combines the strengths of rapid prototyping, release-early-release-often, voluntary participation enforced by the possibility of forks and the global upstream/downstream network with weaknesses in supply chain management, maintenance of license compliance and an at times controversial approach to meritocracy (see 4.5.2). The strengths of SDO include well-established and well-documented formal processes, a strong reputation with industry and policy makers, well-accepted terms of reference and governance frameworks, impactful mechanisms for market signaling and a portfolio of value-added services like publishing and distribution. Weaknesses include institutional inertia, an overwhelming influence of powerful stakeholders, fragmentation, under-defined IPR frameworks and a reliance on outdated revenue streams (see 4.5.3).

In an ideal situation, SDO and FOSS communities cooperate by combining their individual strengths and balancing out their weaknesses in the different phases of the standards development process (see 4.5.1). A number of positive examples of such cooperation exist (ECMAscript at ECMA TC39, C++ standardization at ISO/IEC JTC1) that combine specification in the SDO technical committees with implementation in FOSS, which is also used to validate new features of the standards.

However, there are also situations where the combined weaknesses of SDO and FOSS communities seem to reinforce each other. Both struggle in market segments that are strongly covered by SEP portfolios as well as with the fragmentation evident in both environments that prevents standards and FOSS development from being truly global.

To conclude, our study shows that "SDO and FOSS communities both complement each other and compete for relevance at the same time, but for different aspects of the functions they provide" (see 4.6). By formulating how standardization instruments respond to a need for standardization and cause standardizing effects that lead to the adoption of standards in markets, we show that SDOs represent one standardization instrument and FOSS communities another (see 4.8). The usefulness of leaving implementation to the markets after a standard was promulgated is reduced by the possibility, introduced by FOSS and empirically evident in the growth of "code-first organizations", of collaboratively developing a single joint implementation, which is also seen as an effective method to achieve interoperability (see 4.3.2). We identify cost of change as a key determinant for specification-first versus implementation-first standardization, and highlight that continuous non-differentiating cooperation mostly at FOSS foundations has partially replaced pre-competitive cooperation (see 4.3.3).

6.1.3 On the role of IPR regimes

The pervasive use of general-purpose computers combined with the continuing commoditization of hardware have lead to a situation where devices will typically integrate nearly all basic computing functions like processing, storage, network communication, audio, video, human-machine interfaces and peripheral connections. To build devices, manufacturers require access to all relevant IPR portfolios that cover such functionalities. To avoid a potential anti-commons situation where to acquire all required permissions becomes a prohibitive effort, IPR licensing needs to be streamlined and efficient. Patent pools and FOSS licensing are two approaches the industry has developed for that purpose. Patent pools, both as one-stop-shops for commercial licensing or with OIN focused on facilitating FOSS collaboration, drastically simplify license management by reducing the number of individual patent license contracts and the negotiation effort involved to a minimum. FOSS licenses preempt anti-commons situations for software source code, which makes the ex-post, implicit agreements between authors and consumers essential for the functioning of the upstream/downstream network. The underlying absence of negotiations is a key expectation of FOSS participants, which is one reason why combining FOSS licensing with separate patent or even SEP license requirements has generally not been successful.

The legal compatibility between FRAND and FOSS licensing has long been assumed to be an inhibitor for products that combine both. Our research finds that "the legal compatibility of the licensing conditions is a necessary condition, but not sufficient to establish a successful cooperation between the SDOs and the Open Source software communities."(see

3.4). Embedding FRAND licensed functionality into a product usually requires ex-post negotiations based on, for example, the position of the manufacturer in the supply chain. FOSS licenses communicate expected collaboration models, while in environments that apply FRAND licenses structure commercial relationships. Engagement in SDOs comes with higher barriers to entry compared to FOSS communities. These are examples for detractors to collaboration that make cooperation focusing on FRAND licensed standards between FOSS communities and SDO less probable. Only a few technology areas are dominated by FRAND licensing, primarily telecommunications and, to a lesser extend, multimedia coding. The vast majority of SDOs offer flexible or "toll-free" (see 3.4) licensing regimes that allow for pragmatic compromises to enable collaboration with the wider open source community. At these organisations, there is an ongoing and evolving deeper collaboration between the wider open source community and SDO. Numerous technical standards are implemented in various FOSS technologies, which is supported by a strong overlap of participants in both standards and FOSS development. Especially larger enterprises are typically involved in both. However, governance in SDOs and Open Source communities still differ in key aspects of philosophy and implementation, which remains an obstacle for closer collaboration.

In recent years, FOSS foundations increasingly serve as platforms for horizontal collaboration and consensus building similar to the functions traditionally provided by SDOs. In this carefully regulated environment, standards development continues to serve as a platform for pre-competitive collaboration, where the resulting exchanges in networks of industry experts are a tangible outcome next to the technical standards themselves. This function is also provided by FOSS foundations, where regulatory oversight is further simplified if open governance is applied. Contrary to the assumption that FOSS communities rather manage small to medium size collaborations, while SDO are capable of managing large ones, our research shows that "both SDO and Open Source communities are capable of small to large collaborations (in terms of the number of participants) and small to significant R&D investments" (see 3.5). The converging functions of SDOs and FOSS umbrella organizations offer actors a choice of platforms that previously did not exist. The FOSS community enables a choice for innovators between specification-first, implementation-first, and parallel standardisation that improves the innovativeness of the ICT sector.

6.1.4 On the economics of open source

Chapter 5 returns to the aim of deepening the understanding of as well as building theory about the economics of FOSS in the form of a synthesis of the prior research. Published as a book chapter[20], it is the result of a collaboration of FOSS researchers and practitioners and available under an open access license.[26]

It first positions FOSS as an interdisciplinary topic at the intersection of law, politics and economics. Understanding FOSS holistically instead of as just another way of licensing or just a collaboration methodology is necessary to capture the full scope of societal implications and has often been discounted in the past. The introduction of FOSS into the innovation landscape has wide-ranging implications as a paradigm of creating and sharing knowledge in a digitalized and knowledge-driven society (see 5.1).

Against this context, the article then investigates why FOSS is successful at the societal level, or "why is free software free?" (see 5.2). By explaining how IPR frame one of modern capitalist societies deepest convictions, that creations of the mind are owned by the person that made them, it illustrates the crucial link between copyright-based FOSS licenses and software freedom. As long as that conviction holds, a mechanism like copyright is necessary to declare that an intangible good should be free to use, study, modify and redistribute, or in other words become a public good. In this it disagrees with Benkler who argues that an absence of IP protection would benefit peer production.[11] One consequence of this argument is that while FOSS is assumed to be agnostic to the political system it operates in, the rule of law and the protection of private property is quintessential to the functioning of the wider open source community.

The difficulty to define openness is illustrated against the central role of software freedom in both the Open Source Definition as well as the philosophy of the FSF. The separate dimensions of FOSS licensing and openly governed development processes illustrate that not all FOSS products are created in open community collaboration (see 5.3). The cost of competition in the form of creative destruction as well as the duplicate effort of invention races forces businesses to decide where to compete and where to collaborate. Avoiding the cost of competition at the expense of licensing the outcomes of collaboration freely is a proven strategy to gain access to non-differentiating functionality (see 5.4).

Communities manage community assets including software source code in a model called joint stewardship. Because of regular contributor fluctuation, only a subset of the overall rights holders to the community assets may be engaged in governance at any given time. Other currently non-participating rights holders trust their stewardship (see 5.5). By providing all necessary rights to the stewards up-front and without negotiation, FOSS licensing enables this concept that ensures the continuity of communities. This puts a particular emphasis on defining who is a contributor, what is merit on the FOSS meritocracy and what really makes a FOSS community. By defining FOSS communities as social groups of contributors that participate voluntarily in the production of public information goods, the needs for community governance and community management are explained, as well as how the looming threat of forks serves as a part of the system of checks and balances between the community and the stakeholders in the community product.

The concept of a global upstream/downstream model or upstream/downstream network describes the integration flow along the FOSS supply chain. It is enabled by automatic and transitive licensing which prevents anti-commons situations. FOSS licensing also enables businesses to offer services for components where they are not rights holders. For businesses, engaging in this network requires a choice of either product or service based value propositions, which can be combined with vertical integration or horizontal services offerings. In a commercial context, businesses can benefit from adopting FOSS approaches either by directly generating revenue, by inducing indirect benefits or by reducing cost. To answer why consumers should prefer their product over those of their competitors, businesses need to differentiate their products. Since FOSS is available to any interested user for any purpose, it does not in itself provide a means for differentiation. Businesses compete by complementing the FOSS product for example based on expertise, quality or customer service.

While volunteers contribute a smaller share of the total FOSS development effort compared to corporate ones, they increase the innovativeness and diversity of the FOSS ecosystem and as such represent a core part of it. Their often principled and perfectionist attitude ensures that the wider open source community continues to prioritize charity and virtue beyond just business rationale, as well as inclusive, transparent and consensus-driven conduct. The participation of volunteers contributes to the competitiveness of FOSS, while FOSS licensing combined with open governance inhibits anti-competitive behaviour.

Finally, FOSS licensing is reviewed in the context to recent research about the responses triggered by a price of zero, which frames FOSS development and distribution as social as opposed to market transactions. Fairness and reciprocity are common expectations in social transactions, which helps to explain the hygiene factor character of FOSS license compliance as well as the aversion of the FOSS community to ex-post negotiation, CLAs and contributor asymmetry.

Overall, this article contributes to a deeper understanding of the FOSS ecosystem by formulating in writing concepts that had previously been intuitively understood but not formalized, including the differentiate-or-collaborate "mantra", joint stewardship, the application of FOSS licensing to create unlimited supply, the demands for community governance and community management, FOSS products and services, the difference between vertical integration of software stacks versus horizontal provision of services, as well as community engagement and forks as applications of the conceptual voice-or-exit ultimatum, and others. The key insight that additions to the FOSS pool of technologies contribute positively to the common good establishes a public interest in the productivity and sustainability of the FOSS ecosystem.

6.1.5 On the economic impact of open source

The study "The impact of Open Source software and hardware on technological independence, competitiveness and innovation in the EU economy" provides a comprehensive picture of the commercial uses, costs and benefits of FOSS in Europe. It assesses how the use, promotion and support of FOSS and open hardware will support the EU in achieving its policy goals. The study was commissioned by the European Commission DG CONNECT. It involved a detailed literature review, the performance of several case studies and statistical analyses, and a detailed survey among a representative sample of companies and developers. A strong consistency was observed between the data provided by the various sources consulted, and the data collected specifically for the study.[19] While not included here, the study contributed key insights on the economics and impact of FOSS.

Based on econometric analysis, FOSS developers in the EU contribute significantly to the global FOSS ecosystem. Unlike in the US where corporate contributions are dominated by large enterprises, in the EU SME are most likely to contribute. Companies located in the EU invested about €1 billion in FOSS in 2018, which resulted in an impact on the European economy of between €65 and €95 billion. The FOSS technology pool contributes significantly to the EU's GDP. A 10% increase in contributions from the EU is expected to generate between 0.4 and 0.6% additional EU GDP and more than 600 additional ICT startups per year. The benefits of FOSS to the EU greatly outweigh the costs associated with it, with the lower boundary of the cost-benefit ratio estimated at 1:4. These findings have been evaluated and are supported by a comprehensive stakeholder survey. Case studies additionally investigated open source hardware, where the ecosystem is not yet developed enough to enable quantitative analysis. They revealed a growing community centered around FOSS foundations, with a collaboration that is global rather than focused on the EU only.

The study results support many of the hypotheses and claims made in other parts of this research. For example, the data indicate that contributing companies benefit mainly from openness in particular of standards development and through labour cost savings rather than from generating additional revenue (see for example 5.4). The finding that FOSS moves innovation higher up the value chain is another way of looking at foundational non-differentiating technologies (see 4.2.2). Generally, the study results are coherent with and supportive to this research.

In addition to the core econometric findings, the study reports on a number of open debates and research questions in the wider open source community. It highlights that while openness for software can be relatively well-described against the Open Source Definition, openness of hardware is a question of degree. A similar debate is now taking place at the Open Source Initiative[1] about a definition for open AI.[19, p. 59] The detailed overview of FOSS foundations reveals an ongoing concentration of code-hosting organisations with the Linux Foundation and the Eclipse Foundation as the largest, and that a majority of respondents views foundations as hosts for projects with neutral governance that cannot be controlled by a single organisation.[19, p. 185] Risks of contributing to, adopting and building businesses on top of FOSS are sometimes framed as possible "dark sides to open source". The study discusses some of these arguments including the relationship of large ICT enterprises and communities, the complexity of ensuring license compliance as well as the controversial role of venture-capital funded businesses with FOSS products.[19, p. 215]

International trade policy as well as the international security environment pose questions that the FOSS ecosystem may be unable to answer on its own. Where the wider open source community collaborates across regions in a mostly apolitical fashion, complex geopolitical arguments arise regarding common standards for worker well-being or the respect for human rights. Related to that are discussions about if and to what extend openly sharing dual-use technologies with adversaries may create security risks. These debates relate to the political environment that the wider open source community operates in. The FOSS ecosystem in turn shapes the policy instruments available to regulators (for example by making export controls on foundational technologies like cryptography less viable).

The study concludes with a series of policy recommendations. In particular, it recommends a build-up of EU institutional capacity to support and accelerate open technologies, promoting digital autonomy and technological sovereignty via FOSS, and the development of human and financial capital related to FOSS.

[1] Open Source artificial intelligence (AI): Establishing a common ground (accessed 03/02/2024)

6.2 Limitations

Considering the wide-ranging impact of FOSS on the innovation landscape, this research has only been able to cover a subset of the possible theoretical space. In particular, the application of the research results is limited by the selected scope, the applied methodologies and the time frame in which the research was conducted.

Regarding scope, the three key aspects of governance, IPR regimes and standards development have been selected based on their observable importance for the FOSS ecosystem. Other aspects that may also have relevant impact have been excluded. One of them is the impact of FOSS on software development, in particular software engineering methodologies. A number of FOSS innovations have for all practical matters defined how software is developed today, for example wikis or distributed version control. While the papers refer to those process innovations as "improved methods of collaboration", they represent an important field of study as enablers of global collaboration that is not covered here. Also not included is the impact of FOSS on international innovation and trade policy. Since it is practically impossible to restrict distribution of FOSS technologies at a global scale, there are implications for export regulation, international competition and trade. Especially FOSS technologies sponsored by very large technology enterprises are prone to tensions between open collaboration and trade restrictions. Such topics are not included in this research. Touched upon but not fully covered are impacts of FOSS on the innovation landscape in adjacent fields of technology. For example, the adoption of FOSS licensing and collaboration methods changed R&D, licensing and patenting practises of proprietary, non-free technologies as well. Our research focuses primarily on the areas where FOSS and proprietary technologies interact directly. There is a whole range of impacts on society driven in combination by the adoption of FOSS and the global spread of the internet, including the role of the internet as a space for political debate or the relationship between software freedom and civil liberties. While these aspects are related to the dimension of FOSS as a social movement, they are not deeply covered here.

The methodologies used follow the focus on qualitative analysis and theory building. At the starting point of this research, opposite camps argued with conflicting terminology and theoretical approaches either that IPR regimes need protection from the danger of FOSS, or that FOSS will rescue innovation from the dysfunctional IPR system. This highlighted a need to develop theory that is able to explain observable innovation processes across the whole ICT sector. At the same time, this starting point provided no good foundation for quantitative methodologies. Today and also based on our research, this situation has improved, allowing to study the economic impact of FOSS with quantitative methods. However, the issue of diverging and at times contradicting terminologies and theories has not been fully overcome. It can still be observed, for example, in policy inputs to regulatory initiatives related to SEP or cybersecurity.

Finally, the presented papers are the result of long going research of the FOSS ecosystem, including the participation in two major research projects are EU level (see chapters 3 and [19]). In this period from 2013 to 2023, the wider open source community has continued to evolve as FOSS permeated the ICT sector, with some important changes like the decline of the influence of volunteer-driven communities. However, the focus topics of governance, IPR regimes and standards development continue to be key drivers of the development of the wider open source community, and none of the conclusions have been systematically invalidated during that time. The time frame of this research offered an opportunity to monitor important developments over time, which increases the confidence in the research outcomes. The results should however be interpreted against the period of FOSS adoption in which they where developed.

6.3 Outlook and further research

Even when focusing on entrepreneurial, economic and policy implications and without delving into the enabling effects to software development, it is almost impossible to capture the full impact of FOSS on the landscape of innovation. To illustrate, since FOSS grew beyond the volunteer community to being adopted by industry, no single proprietary technology has been able to conquer a complete market segment again in a way comparable to how Microsoft Windows still dominates the personal computer market. Competition has moved to platform businesses and cloud computing, where the foundational technologies are predominantly FOSS. Such a seismic shift in the fabric of the ICT sector must involve systemic changes to competition, trade, knowledge transfer and industrial R&D. Beyond that, businesses use their involvement in the FOSS ecosystem for public relations, marketing and recruiting. The impact of FOSS on the landscape of innovation really is pervasive.

The eye-catching figure of the economic impact of FOSS on the European economy of between €65 and €95 billion is frequently referenced, and has recently been topped by the estimation of the global demand-side value of FOSS at $8.8 trillion.[67] Beyond estimating the magnitude of the impact, there are important influences at play behind the numbers that warrant further studies.

First, FOSS collaboration introduces implementation-first approaches where there is no time gap between invention and implementation. Both are the same process and happen at the same time in iterative and collaborative fashion. All progress is immediately published and becomes prior art for future iterations. The assumption of a non-zero invention-implementation time gap lies at the heart of the IPR regime, which enables creations of the mind to be transferred from inception to market adoption by granting temporary monopolies that cover the time gap. This research has discovered and illustrated this systematic change FOSS introduces to the IPR framework, however open research questions remain. For example, while copyright terms have been repeatedly extended in recent decades, the product life cycles in ICT have shortened, resulting in a disconnect where the copyright terms are not tailored anymore to help creators bridge the invention-implementation time gap. There is a need for novel research on optimal copyright and patenting terms for ICT technologies. Similarly, limiting the terms of IPR protection in general is justified as a temporary monopoly to cover the invention-implementation time gap gap and to ensure the disclosure of the invention. If this gap does not exist for FOSS and it is disclosed immediately, should IPR protection for FOSS ever expire?

Second, FOSS questions the mutually-exclusive classification of goods into those that are innovative and those that are commodities. Innovative products are what is assumed to drive growth in the ICT sector when they are introduced and then grow in adoption. They become commodities as the market for that product reaches maturity, before it later declines as new products are introduced. FOSS exhibits the fungibility of commodities immediately after contribution, while at the same clearly being new and innovative. Further research on how FOSS is at the same time innovative, state of the art and a commodity and on how that changes product life cycles, competition and innovation promises to be fruitful.

Third, joint stewardship of public goods was evolved in the FOSS community by combining it with a meritocratic selection of maintainers and community leaders as stewards in a dynamic selection process that accounts for contributor fluctuation. Enabled by the automatic and transitive nature of FOSS licensing, the joint stewardship approach should be researched as a role model for the provision of public goods.

Fourth, the practices of the FOSS community show and the theory of peer production postulates that open collaboration combined with open distribution of the outcomes can drive innovation while avoiding some of the negative effects of creative destruction like patent races. Public goods provisioned by government often struggle with keeping up with market developments, while public goods created by FOSS communities showcase exemplary innovativeness. An important future research question therefore is how the model of open collaboration combined with open distribution (as in public money - public code) can be applied in other fields to reap similar innovation benefits.

Fifth, both the wider open source community and the public sector provision public goods. The public sector relies on a monopoly of the state to spend tax revenue. FOSS communities rely on voluntary participation instead of the coordination by a central authority like the state. A political system works best if it achieves its goals when actors act in their own best interest, which they eminently do in the case of FOSS. Also, if the same result can be achieved by either coercion (well-justified, in the case of taxes) or voluntary participation, the later results in a higher degree of freedom for citizens (negative freedom, as in reduced coercion). On the one hand, this offers a choice of either an open collaborative or a centralized governmental approach for the provision of some public goods. On the other hand, the public sector may find itself in the unusual position of a governance taker. International competition and global collaboration mean that even a strong government may not be able to control the how the wider open source community operates. Further research is necessary regarding what approach to the provision of public goods to apply in what situation, how regulation can shape markets in the face of global collaboration, as well as how FOSS and public sector provision can complement each other while avoiding a competition for relevance. Also interesting in this context is that contributions by public sector actors to the FOSS technology pool will be available to everybody for any purpose. As a social transaction, public sector contributions should be considered systematically different from goods and services provided by the public sector in market transactions. More research, in particular political theory building, is necessary about the interaction between public and private sector FOSS contributions and when and how the public sector should be a FOSS contributor.

⋆ ⋆ ⋆

Today, open source is the new normal. FOSS is an integral part of the ICT ecosystem. Digital products build on FOSS components. Open source participation is part of software engineering. The management of open source engagement is engineering management. As the quality, trustworthiness and cybersecurity of software impact markets more and more, the software industry is expected to become increasingly regulated. ICT market regulation is open source regulation, and vice versa. FOSS is a key driver of ICT innovation. Innovation policy is open source policy. In the spirit of open collaboration, all articles in this dissertation are available under open access licenses.

Bibliography

[1] Philippe Aghion et al. "Competition and Innovation: an Inverted-U Relationship". In: *The Quarterly Journal of Economics* 120.2 (May 2005), pp. 701–728.

[2] Fair Standards Alliance. *The importance of maintaining the open source software value proposition*. 2017. URL: https://fair-standards.org/2017/10/23/the-importance-of-maintaining-the-open-source-software-value-proposition/ (visited on 03/02/2024).

[3] O. Almeida, José Oliveira, and José Cruz. "Open Standards and Open Source: Enabling interoperability". In: *International Journal of Software Engineering & Applications* 2.1 (Jan. 2011).

[4] Dan Ariely, Uri Gneezy, and Ernan Haruvy. "Social norms and the price of zero". In: *Journal of Consumer Psychology* 28.2 (2018), pp. 180–191.

[5] Jono Bacon. *The Art of Community: Building the New Age of Participation (Theory in Practice)*. 2nd ed. O'Reilly Media, May 2012.

[6] Malcolm Bain and P McCoy Smith. "Patents and the Defensive Response". In: *Open Source Law, Policy and Practice*. Oxford University Press, Oct. 2022.

[7] Justus Baron and Daniel F. Spulber. "Technology Standards and Standard Setting Organizations: Introduction to the Searle Center Database". In: *Northwestern Law & Econ Research Paper No. 17-16* (Feb. 2018).

[8] Justus Baron et al. *Making the rules – The governance of standard development organizations and their policies on intellectual property rights*. Ed. by N Thumm. Publications Office, 2019.

[9] Rudi Bekkers and Andrew Updegrove. "A Study of IPR Policies and Practices of a Representative Group of Standards Setting Organizations Worldwide". In: *SSRN Electronic Journal* (Jan. 2012).

[10] Daniel Bell. "On meritocracy and equality". In: *National Affairs* 29 (1972), pp. 29–68.

[11] Yochai Benkler. “Coase’s Penguin, or, Linux and "The Nature of the Firm"”. In: *The Yale Law Journal* 112.3 (Dec. 2002), pp. 369+.

[12] Mark Bevir. *Governance - a very short introduction*. Oxford University Press, 2012.

[13] Knut Blind. “An economic analysis of standards competition: The example of the ISO ODF and OOXML standards”. In: *Telecommunications Policy* 35.4 (2011), pp. 373–381.

[14] Knut Blind. “Driving forces for standardization at standardization development organizations”. In: *Applied Economics* 34.16 (2002), pp. 1985–1998.

[15] Knut Blind and Mirko Boehm. *The Relationship Between Open Source Software and Standard Setting*. 2019.

[16] Knut Blind, Mirko Böhm, and Nikolaus Thumm. “Open Source Software in Standard Setting: The Role of Intellectual Property Right Regimes”. In: *Open Source Law, Policy and Practice*. Oxford University Press, Oct. 2022.

[17] Knut Blind and Axel Mangelsdorf. “Motives to standardize: Empirical evidence from Germany”. In: *Technovation* 48-49 (Feb. 2016), pp. 13–24.

[18] Knut Blind et al. *Study on the interaction between standardisation and intellectual property rights*. Fraunhofer ISI, 2002.

[19] Knut Blind et al. *The impact of open source software and hardware on technological independence, competitiveness and innovation in the EU economy – Final study report*. Publications Office, 2021.

[20] Mirko Boehm. “Economics of Open Source”. In: *Open Source Law, Policy and Practice*. Oxford University Press, Oct. 2022.

[21] Mirko Boehm. “The emergence of governance norms in volunteer driven open source communities”. In: *Journal of Open Law, Technology & Society* 11.1 (2019), pp. 3–39.

[22] Mirko Boehm and Davis Eisape. “Standard setting organizations and open source communities: Partners or competitors?” In: *First Monday* 26.7 (June 2021).

[23] Jutta Bolt et al. “Rebasing ‘Maddison’: new income comparisons and the shape of long-run economic development”. In: *GGDC Research Memorandum* 174 (2018).

[24] Anu Bradford. *The Brussels effect: How the European Union rules the world*. Oxford University Press, USA, 2020.

[25] Amanda Brock. “Business and Revenue Models and Commercial Agreements”. In: *Open Source Law, Policy and Practice*. Oxford University Press, Oct. 2022.

[26] Amanda Brock, ed. *Open Source Law, Policy and Practice*. Oxford University Press, Oct. 2022.

[27] Hilary Carter and Jessica Groopman. *Diversity, Equity, and Inclusion in Open Source: Exploring the Challenges and Opportunities to Create Equity and Agency Across Open Source Ecosystems*. The Linux Foundation, Dec. 2021.

[28] Henry Chesbrough. *Open business models: How to thrive in the new innovation landscape*. Harvard Business Press, 2006.

[29] Henry Chesbrough, Wim Vanhaverbeke, and Joel West. *Open innovation: Researching a new paradigm*. Oxford University Press on Demand, 2006.

[30] Pamela Chestek. “Trademarks”. In: *Open Source Law, Policy and Practice*. Oxford University Press, Oct. 2022.

[31] Jamie Clark. *OASIS at ITU/NGMN: Convergence, Collaboration and Smart Shopping in Open Standards and Open Source*. 2017. URL: https://www.slideshare.net/JamieClark1/oasis-at-itungmn-convergence-collaboration-and-smart-shopping-in-open-standards-and-open-source-81474699 (visited on 03/02/2024).

[32] Ronald H. Coase. “The Nature of the Firm”. In: *Economica*. New Series 4.16 (Nov. 1937), pp. 386–405.

[33] Stefano Comino, Fabio M Manenti, and Nikolaus Thumm. “The role of patents in information and communication technologies: A survey of the literature”. In: *Journal of Economic Surveys* 33.2 (2019), pp. 404–430.

[34] Robert G. Cooper. “Agile-Stage-Gate Hybrids”. In: *Research-Technology Management* 59.1 (2016), pp. 21–29.

[35] Jonathan Corbet and Greg Kroah-Hartman. *2017 Linux Kernel Development Report*. Tech. rep. Linux Foundation, 2017.

[36] Jonathan Corbet and Greg Kroah-Hartman. *Linux Kernel Development, 25th Anniversary Edition*. Tech. rep. Linux Foundation, 2016.

[37] Daniel Curto-Millet and Alberto Corsín Jiménez. “The sustainability of open source commons”. In: *European Journal of Information Systems* 32.5 (2023), pp. 763–781.

[38] Carlo Daffara. “Estimating the Economic Contribution of Open Source Software to the European Economy”. In: *The First Open-Forum Academy Conference Procedings*. Ed. by Shane Coughlan. OpenForum Europe, 2012.

[39] Chris DiBona and Sam Ockman. *Open sources: Voices from the open source revolution*. "O’Reilly Media, Inc.", 1999.

[40] Avinash Dixit and Mancur Olson. “Does voluntary participation undermine the Coase Theorem?” In: *Journal of Public Economics* 76.3 (2000), pp. 309–335.

[41] Nadia Eghbal. *Roads and bridges: the unseen labor behind our digital infrastructure.* 2018.

[42] Harvard Business Essentials. “Strategy: Create and implement the best strategy for your business”. In: *Boston, Massachusetts: Harvard Business School* (2005).

[43] ETSI. *Open Source NFV Management and Orchestration (MANO).* 2012. URL: http://www.etsi.org/technologies-clusters/technologies/nfv/open-source-mano (visited on 03/02/2024).

[44] Amitai Etzioni. “Two Approaches to Organizational Analysis: A Critique and a Suggestion”. In: *Administrative Science Quarterly* 5.2 (1960), pp. 257–278.

[45] Free Software Foundation Europe. *Why is FRAND bad for Free Software?* URL: https://fsfe.org/activities/os/why-frand-is-bad-for-free-software.en.html (visited on 03/02/2024).

[46] European Commission. *Communication from the Commission to the European Parliament, the Council and the European Economic and Social Committee: Setting out the EU approach to Standard Essential Patents.* Publications Office of the European Union, Nov. 2017.

[47] European Commission. *European Interoperability Framework.* Publications Office of the European Union, 2017.

[48] European Commission. *Implementing FRAND standards in Open Source: Business as usual or mission impossible? Report on the joint European Commission and European Patent Office conference "Implementing FRAND standards in Open Source" on 22 November 2012.* 2016. URL: https://ec.europa.eu/docsroom/documents/15601 (visited on 03/02/2024).

[49] European Commission and Directorate-General for Internal Market, Industry, Entrepreneurship and SMEs. *User guide to the SME definition.* Publications Office of the European Union Luxembourg, 2020.

[50] European DIGITAL SME Alliance. *CEN-CENELEC Initiative on Digital Transformation: an opportunity for SMEs.* Oct. 2017. URL: https://www.digitalsme.eu/cen-cenelec-initiative-digital-transformation-opportunity-smes/ (visited on 03/02/2024).

[51] Karl Fogel. *Producing Open Source Software: How to Run a Successful Free Software Project.* O’Reilly Media, Inc., Oct. 2005.

[52] Erwin Folmer, Michael van Bekkum, and Jack Verhoosel. "Strategies for using international domain standards within a national context: The case of the Dutch temporary staffing industry". In: *Innovations for Digital Inclusions, 2009*. IEEE, Sept. 2009.

[53] The Linux Foundation. *Linux Foundation Launches OpenTofu: A New Open Source Alternative to Terraform*. 2023. URL: https://www.linuxfoundation.org/press/announcing-opentofu (visited on 03/02/2024).

[54] Free Software Foundation. *What is Free Software?* 1996. URL: https://www.gnu.org/philosophy/free-sw.html (visited on 03/02/2024).

[55] Free Software Foundation Europe. *Public Money, Public Code!* 2018. URL: https://publiccode.eu/ (visited on 03/02/2024).

[56] Jonas Gamalielsson et al. "On organisational influences in software standards and their open source implementations". In: *Information and Software Technology* 67 (2015), pp. 30–43.

[57] Urs Gasser and John Palfrey. "When and How ICT Interoperability Drives Innovation". In: *Berkman Publication Series* (Nov. 2007).

[58] Joshua Gay. *The Principles of Community-Oriented GPL Enforcement*. Sept. 2015. URL: https://www.fsf.org/licensing/enforcement-principles (visited on 03/02/2024).

[59] Rishab A Ghosh. "An economic basis for open standards". In: *Opening Standards: The Global Politics of Interoperability, MIT Press, Cambridge, MA* (2011), pp. 75–96.

[60] Garett Hardin. "The Tragedy of the Commons". In: *Science* 162.3859 (Dec. 1968), pp. 1243–1248.

[61] Russell Hardin. *Collective Action*. RFF Press, June 1982.

[62] Michael A Heller. "The tragedy of the anticommons: property in the transition from Marx to markets". In: *Harvard law review* (1998), pp. 621–688.

[63] Armijn Hemel and Shane Coughlan. *Practical GPL Compliance*. Linux Foundation, 2017.

[64] F. Herzberg, B. Mausner, and B. B. Snyderman. *The Motivation to Work*. Transaction Publishers, Jan. 1993.

[65] A. O. Hirschman. *Exit, Voice, and Loyalty: Responses to Decline in Firms, Organizations, and States*. New edition. Harvard University Press, July 1970.

[66] Albert O. Hirschman. *The Passions and the Interests*. 20 Anv Sub. Princeton University Press, Jan. 1997.

[67] Manuel Hoffmann, Frank Nagle, and Yanuo Zhou. “The Value of Open Source Software”. In: *Harvard Business School Strategy Unit Working Paper* 24-038 (Jan. 2024).

[68] G. Houben, K. Lenie, and K. Vanhoof. “A knowledge-based SWOT-analysis system as an instrument for strategic planning in small and medium sized enterprises”. In: *Decision Support Systems* 26.2 (1999), pp. 125–135.

[69] Paula Hunter and Stephen Walli. “The Rise and Evolution of the Open Source Software Foundation”. In: *Journal of Open Law, Technology & Society* 5.1 (2013), pp. 31–42.

[70] ISO. *ISO 2017 Annual Report*. 2017. URL: https://www.iso.org/files/live/sites/isoorg/files/about%20ISO/annual_reports/en/annual_report_2017_en.pdf (visited on 03/02/2024).

[71] ITU-T. *V.24: List of definitions for interchange circuits between data terminal equipment (DTE) and data circuit-terminating equipment (DCE)*. Feb. 2000. URL: https://www.itu.int/rec/T-REC-V.24-200002-I/en (visited on 03/02/2024).

[72] Rhys Jenkins. “Corporate Codes of Conduct - Self-Regulation in a Global Economy”. In: *Technology, Business and Society Programme, United Nations Research Institute for Social Development* (Apr. 2001).

[73] Omar Johnny, Marc Miller, and Mark Webbink. “Copyright in Open Source Software - Understanding the Boundaries”. In: *Journal of Open Law, Technology & Society* 2.1 (2010), pp. 13–38.

[74] David J Kappos and Miling Y Harrington. “The Truth about OSS-FRAND: By All Indications, Compatible Models in Standards Settings”. In: *Colum. Sci. & Tech. L. Rev.* 20 (2018), p. 242.

[75] Andrew Katz. “Open Hardware”. In: *Open Source Law, Policy and Practice*. Oxford University Press, Oct. 2022.

[76] Michael L. Katz and Carl Shapiro. “Technology Adoption in the Presence of Network Externalities”. In: *The Journal of Political Economy* 94.4 (1986), pp. 822–841.

[77] Richard Kemp. “Towards Free/Libre Open Source Software (“FLOSS”) Governance in the Organisation”. In: *Journal of Open Law, Technology & Society* 1.2 (2009), pp. 61–72.

[78] Greg Kroah-Hartman et al. *Linux Kernel Community Enforcement Statement*. Oct. 2017. URL: http://kroah.com/log/blog/2017/10/16/linux-kernel-community-enforcement-statement/ (visited on 03/02/2024).

[79] Liz Laffan. “A New Way of Measuring Openness: The Open Governance Index”. In: *Technology Innovation Management Review* 2 (2012), pp. 18–24.

[80] Karim Lakhani and Robert G. Wolf. “Why Hackers Do What They Do: Understanding Motivation and Effort in Free/Open Source Software Projects”. In: *Social Science Research Network Working Paper Series* (Sept. 2003).

[81] Adrienn Lawson and Stephen Hendrick. *Global Spotlight 2023: Survey-based insights into the global landscape of open source trends, sustainability challenges, and growth opportunities.* The Linux Foundation, Oct. 2023.

[82] Josh Lerner and Jean Tirole. “Standard-Essential Patents”. In: *Journal of Political Economy* 123 (June 2015), pp. 547–586.

[83] Josh Lerner and Jean Tirole. “The economics of technology sharing: Open source and beyond”. In: *Journal of Economic Perspectives* 19.2 (2005), pp. 99–120.

[84] Josh Lerner and Jean Triole. “The Simple Economics of Open Source”. In: *National Bureau of Economic Research Working Paper Series* (Mar. 2000), pp. 7600+.

[85] Lawrence Lessig. *Code: And other laws of cyberspace.* ReadHowYouWant. com, 2009.

[86] Jingze Li. “Intellectual property licensing tensions in incorporating open source into formal standard setting context — The case of Apache V.2 in ETSI as a start”. In: Nov. 2017, pp. 1–8.

[87] Jeff Licquia and Amanda McPherson. *A $5 Billion Value: Estimating the Total Development Cost of Linux Foundation’s Collaborative Projects.* Tech. rep. https://www.linux.com/publications/estimating-total-development-cost-linux-foundations-collaborative-projects. The Linux Foundation.

[88] Johan Linåker et al. “How Firms Adapt and Interact in Open Source Ecosystems: Analyzing Stakeholder Influence and Collaboration Patterns”. In: *Requirements Engineering: Foundation for Software Quality: 22nd International Working Conference, REFSQ 2016, Gothenburg, Sweden, March 14-17, 2016, Proceedings.* Ed. by Maya Daneva and Oscar Pastor. Cham: Springer International Publishing, 2016, pp. 63–81.

[89] Jilayne Lovejoy. “Contributor Agreements”. In: *Open Source Law, Policy and Practice.* Oxford University Press, Oct. 2022.

[90] Björn Lundell and Jonas Gamalielsson. “On the potential for improved standardisation through use of open source work practices in different standardisation organisations: How can open source-projects contribute to development of IT-standards?” In: *The 22nd EURAS Annual Standardisation Conference, Berlin, Germany, June 28-30, 2017*. Verlag Mainz. 2017, T137–T155.

[91] Björn Lundell, Jonas Gamalielsson, and Andrew Katz. “On implementation of Open Standards in software : To what extent can ISO standards be implemented in open source software?” In: *International Journal of Standardization Research* 13.1 (2015), pp. 47–73.

[92] Günther Luxbacher. *DIN von 1917 bis 2017*. Beuth Verlag, 2017.

[93] Olson Mancur. *The Logic of Collective Action: Public Goods and the Theory of Groups*. Revised. Harvard economic studies, v. 124. Harvard University Press, Jan. 1965.

[94] Catharina Maracke. “Free and Open Source Software and FRAND-based patent licenses: How to mediate between Standard Essential Patent and Free and Open Source Software”. In: *The Journal of World Intellectual Property* 22.3-4 (2019), pp. 78–102.

[95] Heather Meeker. *Patrick McHardy and copyright profiteering*. Aug. 2017. URL: https://opensource.com/article/17/8/patrick-mchardy-and-copyright-profiteering (visited on 03/02/2024).

[96] Michael Meuser and Ulrike Nagel. “ExpertInneninterviews—vielfach erprobt, wenig bedacht”. In: *Das Experteninterview*. Springer, 2002, pp. 71–93.

[97] Iain Mitchell and Stephen Mason. “Compatibility Of The Licensing Of Embedded Patents With Open Source Licensing Terms”. In: *Journal of Open Law, Technology & Society* 3.1 (2011), pp. 25–58.

[98] Frank Nagle. “Government Technology Policy, Social Value, and National Competitiveness”. In: *Harvard Business School Strategy Unit Working Paper* 19-103 (2019), pp. 1–50.

[99] Frank Nagle. “Open source software and firm productivity”. In: *Management Science* (2018).

[100] Frank Nagle et al. *Census II of Free and Open Source Software - Application Libraries*. The Linux Foundation and The Laboratory for Innovation Science at Harvard, Mar. 2022.

[101] Frank Nagle et al. *Vulnerabilities in the Core: Preliminary Report and Census II of Open Source Software*. The Linux Foundation & The Laboratory for Innovation Science at Harvard, 2020.

[102] OASIS OpenDocument Technical Committee. *OASIS Open Document Format for Office Applications*. 2009. URL: https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=office (visited on 03/02/2024).

[103] OECD. *Demand-side Innovation Policies*. 2011, p. 188.

[104] OECD and Eurostat. *Oslo Manual 2018*. 2019, p. 256.

[105] Open Container Initiative. *Mission*. 2016. URL: https://www.opencontainers.org/ (visited on 03/02/2024).

[106] Open Source Initiative. *The Open Source Definition*. 2006. URL: https://opensource.org/%20osd/ (visited on 03/02/2024).

[107] OpenForum Europe. *Standards and Open Source - Bringing them together*. Nov. 2017. URL: https://ec.europa.eu/digital-single-market/en/news/standards-and-open-source-bringing-them-together (visited on 03/02/2024).

[108] opensource.com. *The open source way*. 2018. URL: https://opensource.com/open-source-way (visited on 03/02/2024).

[109] Cailean Osborne, Mirko Boehm, and Ana Jimenez Santamaria. *The European Public Sector Open Source Opportunity: Challenges and Recommendations for Europe's Open Source Future*. The Linux Foundation, Sept. 2023.

[110] Elinor Ostrom. "Collective Action and the Evolution of Social Norms". In: *The Journal of Economic Perspectives* 14.3 (2000), pp. 137–158.

[111] Paul Parker-Johnson and Tim Doiron. *Succeeding on an Open Field: The Impact of Open Source Technologies on the Communication Service Provider Ecosystem*. 2018.

[112] Chryssoula Pentheroudakis, Justus Baron, and Nikolaus Thumm. *Licensing Terms of Standard Essential Patents: A Comprehensive Analysis of Cases*. Scientific analysis or review, Policy assessment. Luxembourg (Luxembourg), 2017.

[113] Bruce Perens. *On Usage of The Phrase "Open Source"*. Sept. 2017. URL: https://perens.com/2017/09/26/on-usage-of-the-phrase-open-source/ (visited on 03/02/2024).

[114] Simon Phipps. *Open source procurement: Copyrights*. Feb. 2011. URL: https://opensource.com/law/11/2/open-source-procurement-copyrights (visited on 03/02/2024).

[115] Simon Phipps. *Why Legal Issues are the Wrong Lens to understand the Open Source and FRAND issue*. 2019. URL: https://openforumeurope.org/wp-content/uploads/2019/03/OFA_-_Opinion_Paper_-_Simon_Phipps_-_OSS_and_FRAND-1.pdf (visited on 03/02/2024).

[116] Lydia Pintscher, ed. *20 Years of KDE: Past, Present and Future*. KDE e.V., Aug. 2016.

[117] Tim C. Pohlmann. "Attributes and Dynamic Development Phases of Informal ICT Standards Consortia". In: *Social Science Research Network Working Paper Series* (Mar. 2013).

[118] Karl Raimund Popper. *The open society and its enemies*. Routledge & Kegan Paul, 1957.

[119] Eric S Raymond and Guy L Steele. *The new hacker's dictionary*. Mit Press, 1996.

[120] Eric S. Raymond. *The Cathedral and the Bazaar: Musings on Linux and Open Source by an Accidental Revolutionary (O'Reilly Linux)*. O'Reilly, Oct. 1999.

[121] Red Hat. *Increasing stability and predictability in Open Source license compliance: Providing a Fair Chance to Correct Mistakes*. Nov. 2017. URL: https://www.redhat.com/en/about/gplv3-enforcement-statement (visited on 03/02/2024).

[122] Dirk Riehle. "The single-vendor commercial open course business model". In: *Information Systems and E-Business Management* (Nov. 2010), pp. 1–13.

[123] Dirk Riehle et al. "Paid vs. Volunteer Work in Open Source". In: *System Sciences (HICSS), 2014 47th Hawaii International Conference on*. IEEE, Jan. 2014, pp. 3286–3295.

[124] Jeffrey A. Roberts, Il-Horn Hann, and Sandra A. Slaughter. "Understanding the Motivations, Participation, and Performance of Open Source Software Developers: A Longitudinal Study of the Apache Projects". In: *Management Science* 52.7 (2006), pp. 984–999.

[125] Gregorio Robles and Jesús M. González-Barahona. "A Comprehensive Study of Software Forks: Dates, Reasons and Outcomes". In: *Open Source Systems: Long-Term Sustainability*. Ed. by Imed Hammouda et al. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 1–14.

[126] Mikael Rogers. *Growing a contributor base in modern open source*. 2016. URL: https://opensource.com/life/16/5/growing-contributor-base-modern-open-source (visited on 03/02/2024).

[127] Lawrence Rosen. *Defining Open Standards*. URL: http://www.rosenlaw.com/pdf-files/DefiningOpenStandards.pdf (visited on 03/02/2024).

[128] Karen Sandler. “Foundations and Other Organisations”. In: *Open Source Law, Policy and Practice*. Oxford University Press, Oct. 2022.

[129] Jan-Felix Schrape. “Open-source projects as incubators of innovation: From niche phenomenon to integral part of the industry”. In: *Convergence* 25.3 (2019), pp. 409–427.

[130] Joseph A Schumpeter. *Capitalism, socialism and democracy*. Routledge, 2010.

[131] Sonali K. Shah. “Motivation, Governance, and the Viability of Hybrid Forms in Open Source Software Development”. In: *Management Science* 52.7 (2006), pp. 1000–1014.

[132] Kristina Shampanier, Nina Mazar, and Dan Ariely. “Zero as a special price: The true value of free products”. In: *Marketing science* 26.6 (2007), pp. 742–757.

[133] Carl Shapiro and Hal R. Varian. *Information Rules: A Strategic Guide to the Network Economy*. 1ST. Harvard Business Review Press, Nov. 1998.

[134] Dong-Hee Shin, Hongbum Kim, and Junseok Hwang. “Standardization revisited: A critical literature review on standards and innovation”. In: *Computer Standards & Interfaces* 38 (2015), pp. 152–157.

[135] P McCoy Smith. “Copyright, Contract, and Licensing in Open Source”. In: *Open Source Law, Policy and Practice*. Oxford University Press, Oct. 2022.

[136] Diomidis Spinellis and Georgios Gousios. *Beautiful Architecture: Leading Thinkers Reveal the Hidden Beauty in Software Design*. O’Reilly Media, Inc., Jan. 2009.

[137] Steven M. Spivak and F. Cecil Brenner. *Standardization Essentials: Principles and Practises*. New York: CRC Press, 2001.

[138] Daniel F. Spulber. “Consumer coordination in the small and in the large: Implications for antitrust in markets with network effects”. In: *Journal of Competition Law & Economics* 4.2 (2008), pp. 207–262.

[139] Robert E. Stake. *Qualitative Research: Studying How Things Work*. 1st ed. New York: The Guilford Press, Mar. 2010.

[140] Richard M. Stallman and Lawrence Lessig. *Free software, free society: selected essays of Richard M. Stallman*. SoHo Books, 2010.

[141] Fernando F. Suárez and James M. Utterback. “Dominant designs and the survival of firms”. In: *Strategic Management Journal* 16.6 (1995), pp. 415–430.

[142] The Apache Software Foundation. *Briefing: The Apache Way*. 2024. URL: https://www.apache.org/theapacheway/ (visited on 03/02/2024).

[143] The International Churchill Society. *The worst form of government*. 2016. URL: https://winstonchurchill.org/resources/quotes/the-worst-form-of-government/ (visited on 03/02/2024).

[144] Jean Tirole. *Economics for the common good*. Princeton University Press, 2017.

[145] Valerio Torti. *Intellectual Property Rights and Competition in Standard Setting: Objectives and Tensions*. Routledge, 2015.

[146] Edna Ullmann-Margalit. *The emergence of norms*. Clarendon Press, 1977.

[147] Peter Ullrich. “Das explorative ExpertInneninterview: Modifikationen und konkrete Umsetzung der Auswertung von ExpertInneninterviews nach Meuser-Nagel”. In: *Die Transformation des Politischen: Analysen, Deutungen und Perspektiven; siebentes und achtes DoktorandInnenseminar der Rosa-Luxemburg-Stiftung* 66 (2006). Ed. by Tim Engartner, Diana Kuring, and Thorsten Teubl, pp. 100–109.

[148] Andy Updegrove. *Licensing Standards that Include Code: Heads or Tails?* 2015. URL: https://www.consortiuminfo.org/intellectual-propery/intellectual-property-rights/licensing-standards-that-include-code-heads-or-tails/ (visited on 03/02/2024).

[149] Hal R. Varian. “Buying, Sharing and Renting Information Goods”. In: (1997), pp. 473–488.

[150] Hal R. Varian. “Pricing Information Goods”. In: (1995). Ed. by Hal R. Varian.

[151] Bettina Von Stamm. *Managing innovation, design and creativity*. John Wiley & Sons, 2008.

[152] Huibert de Vries, Henk de Vries, and Ilan Oshri. *Standards Battles in Open Source Software: The Case of Firefox*. London: Palgrave Macmillan, 2008.

[153] W3C. *Web Standards*. 2020. URL: https://www.w3.org/standards/ (visited on 03/02/2024).

[154] Steven Weber. *The Success of Open Source*. Harvard University Press, Apr. 2004.

[155] Jim Whitehurst. *The open organization: Igniting passion and performance*. Harvard Business Review Press, 2015.

[156] Paul Moritz Wiegmann, Henk J. de Vries, and Knut Blind. “Multi-mode standardisation: A critical review and a research agenda”. In: *Research Policy* 46.8 (2017), pp. 1370–1386.

[157] Anthony Williams. *Standing Together on Shared Challenges: Report on the 2023 Open Source Congress*. The Linux Foundation, Dec. 2023.

[158] World Economic Forum. *Digital Transformation of Industries: Societal Implications*. World Economic Forum, 2016.

[159] World Intellectual Property Organization. *Berne Convention for the Protection of Literary and Artistic Works*. 1967. URL: https://treaties.un.org/pages/showdetails.aspx?objid=0800000280115ec9 (visited on 03/02/2024).

[160] Robert K. Yin. *Case Study Research: Design and Methods*. 3rd. New York: SAGE Publications, Inc, Dec. 2003.

[161] Michael Dunlop Young. *The rise of the meritocracy, 1870-2033: The new elite of our social revolution*. Vol. 85. Random House, 1959.

[162] Yue Yu et al. “Exploring the Patterns of Social Behavior in GitHub”. In: *Proceedings of the 1st International Workshop on Crowd-Based Software Development Methods and Technologies*. CrowdSoft 2014. Hong Kong, China: Association for Computing Machinery, 2014, pp. 31–36.

Glossary

ad-hoc coordination

Ad-hoc coordination happens in social groups that work towards a common goal without any formal governance structure. It is usually only viable in small groups or for efforts with limited scope. 23

bike-shedding

Bike-shedding describes the habit of bringing up tangential arguments in a debate that distract from the issue at hand. It is named after the question of which colour to paint the bike shed when the debate is about whether or not to build one. The use of the term in the wider open source community originated in the BSD community in 1999. It has been observed in KDE and many other communities. 25, 55, 64, 66, 68

code of conduct

A code of conduct describes the behavioral norms and rules that community contributors are expected to adhere to. Once adopted, a written code of conduct is part of the explicit governance norms of a community. 16, 17, 29, 54, 57

community

A FOSS community consists of the network of participants that contribute to the creation of specific FOSS software products, predominantly computer programs, in a collaborative peer productions process based on voluntary participation. Communities may be set up implicitly or explicitly as unincorporated associations, as for-profit or not-for-profit legal entities or under FOSS umbrella organizations. 194, 220

community composition

Community composition describes how the contributors to a community are comprised of individual volunteers, organisations, employed staff and other constituencies. It is assumed that community composition influences governance norms. Hybrid communities composed of a mix of businesses, individual volunteers and possibly staff are most common. 2, 6, 25, 27, 72, 222

community working group

The KDE community working group is the only formal conflict resolution mechanism the KDE community has in place. “The long-term goal of the Community Working Group is to help to maintain a friendly and welcoming KDE community, thereby ensuring KDE remains a great project enjoyed by all contributors and users.”[2] 54, 57

continuous non-differentiating cooperation

Continuous non-differentiating cooperation is a collaboration model implemented in FOSS communities and facilitated by FOSS foundations. It enables otherwise competing market actors to continuously cooperate to develop a common software stack that serves as basic, non-differentiating technology prerequisite to products that combine free and proprietary software. In contrast to pre-competitive cooperation on the development of proprietary, differentiating products, anti-trust concerns are not relevant in the continous-non-differentiating cooperation model since collusion is impossible if the results are immediately available to the general public and the development process is generally open for participation to all interested parties. The possibility of forks limits the control individual entities can exercise over the development process. 107, 108, 132, 137, 140, 143, 176, 191

contributor asymmetry

The absence of contributor symmetry, for example in single-vendor open source businesses that use CLAs to centralize control over the project. 195, *see* contributor symmetry

contributor symmetry

Contributor symmetry describes the normative expectation that all contributors have equal access to the development and governance processes fo a community and that the community product is managed under joint stewardship. Under contributor symmetry, contributors usually retain the copyright to their code contributions, so that no single entity is able to exercise control over the community product. 184

[2] `https://ev.kde.org/workinggroups/cwg/` (accessed 03/02/2024)

corporate contributor

Corporate contributors participate in FOSS communities as organizations with the resources available to them. Organizations typically contribute financially to projects, for example by way of membership fees, as well as by assigning staff to make technical contributions to a project. It is however the organization that decides to contribute, usually following business rationale as opposed to the individual motivation of volunteer contributors. 2, 88, 138, 186, 222, 229

Cyber Resilience Act

The "regulation on horizontal cybersecurity requirements for products with digital elements" or "Cyber Resilience Act" establishes cybersecurity requirements for products made available in the EU internal market. 4, 226, 231

Debian Social Contract

"Debian Social Contract" refers to the Debian developers' commitment towards the free software community. The Debian Social Contract is a 'moral agenda' which is based on the Debian Free Software Guidelines according to which Debian promises to be entirely free, provide the best work, not try to cover problems, give priority to users and free software and to make it possible for the software to be used with non-free software. The commitments of the Debian Social Contract have later been adopted into the Open Source Definition.[3] 183

digital sovereignty

Digital sovereignty, also referred to as technological sovereignty, describes a notion of strategic autonomy in the development and operation of digital products. The EU digital policy mentions data protection, cybersecurity and artificial intelligence as aspects of digital sovereignty. 3

Eclipse Foundation

The Eclipse Foundation is a not-for-profit industry association (US 501(c)(6)) that provides a global community of individuals and organizations with a mature, scalable, and commercially focused environment for collaboration and innovation. 106, 124, 220, 228

Ecma

Ecma, formerly the European Computer Manufacturers Association (ECMA) is a SDO for information and communication systems. 190,

[3] https://www.debian.org/social_contract (accessed 03/02/2024)

231

excludable

A good is excludable if access to it can be controlled so that a person can be prevented from consuming it without permission. FOSS licences make software a non-excludable good. 18

fair, reasonable and non-discriminatory

Licensing under fair, reasonable and non-discriminatory terms is a voluntary commitment some SDO request from a patent owner that participates in standards development. What FRAND commitments entail exactly is not normalized. It is left to negotiation between market participants and usually confidential. This makes it difficult to ensure non-discrimination and is considered to disadvantage FOSS communities, which makes the FRAND concept controversial.[45] 9, 75–82, 86, 87, 89–91, 102, 118, 119, 128, 131, 137, 191, 192, 226, 232, 234

fiduciary licence agreement

With a fiduciary licence agreement, a contributor delegates some rights to their contributions to a trustee, usually a community organisation. FLAs are a type of contributor agreement, however they aim to avoid the concentration of rights in the hands of a single commercial entity. 232

fork

A fork is a split of the development of a FOSS product into two new communities. Forks introduce duplication of efforts and other downsides and are usually avoided unless there is strong disagreement over community governance. Prominent forks are LibreOffice (forked from OpenOffice), MariaDB (forked from or) MySQL or Nextcloud (forked from ownCloud). The term fork is sometimes also used in the sense of branches for experimental development efforts or on Github for a copy of a repository. 25

foundation

A FOSS foundation is a legal entity that hosts one or more projects as a governance body. In the FOSS context, a foundation can also act as an umbrella organization such as seen in the Linux Foundation, Eclipse Foundation or Wikimedia Foundation to host multiple, different types of projects. 3, 84, 226

free and open source software

The term free and open source software refers to software that is distributed under a license which complies with the Open Source Definition or respectively the definition of free software provided by the FSF. The Open Source Initiative is the steward that approves licenses for being compliant with the Open Source Definition. Free and open source software is made available to everybody under a license that gives the user the freedom to run, copy, distribute, study, change and improve the software. "'Free software' is a matter of liberty, not price".[140] The FSF and the Open Source Initiative maintain lists of licenses[4] that provide at least those "four essential freedoms" to recipients of the software. The terms of all licenses that provide these terms turn the product into a public good and also ensure that the product itself will continue to be freely licensed. This effect may or may not extend to derivative works and future versions, resulting in the classification of the licenses into reciprocal and permissive categories. 1–13, 16–23, 25–28, 30–33, 37–40, 42, 44, 50, 53, 57, 64, 66, 71, 72, 75–107, 110–127, 131–145, 154, 182–202, 217–221, 223–229, 232, 234

Free Software Foundation Europe

The Free Software Foundation Europe "is a charity that empowers users to control technology".[5] 16, 25, 29, 43–50, 64, 65, 67, 70, 145, 221, 224, 232

Freedom Task Force

The *Freedom Task Force* was an intiative of FSFE to to help programmers properly set up and organise projects legally, as well as educate companies to understand how the GPL works.[6] The Freedom Task Force was created in 2006. 44

FSF

"The Free Software Foundation (FSF) is a nonprofit with a worldwide mission to promote computer user freedom."[7] 9, 27, 43, 45, 125, 148, 153, 174, 193, 221

governance

In the context of FOSS, governance describes the totality of implicit and explicit behavioral norms, codes and processes that regulate the relationship between contributors and the community as a whole. 1, 3–6, 8, 9, 13, 18–24, 26–31, 183–188, 190, 192, 194, 195, 197–199, 217

[4] `https://www.gnu.org/licenses/license-list.html` (accessed 03/02/2024), `https://opensource.org/licenses` (accessed 03/02/2024)

[5] `https://fsfe.org/about/about.en.html` (accessed 03/02/2024)

[6] `https://fsfe.org/activities/self/self.en.html` (accessed 03/02/2024)

[7] `http://www.fsf.org/` (accessed 03/02/2024)

governance taker

A governance *taker* is a contributor who generally must accept the prevailing governance norms in a community, usually because the contributor is one of many in an overall healthy community of many independent participants. The terminology is analogous to the idea of price makers and price takers in markets. 201

governing board

A governing board is appointed by the project stakeholders and responsible for project steering. Depending on community composition and the community's governance norms, the stakeholders may be a mix of contributors, funding member organizations and other entities. The responsibilities of the governing board usually exclude technical project steering, which is performed by a TSC. 228

hybrid community

The term "hybrid community" references a community composition with a mix of volunteer contributors and corporate contributors. 2, 26, 187

ICT

"Information and communication technology" describes the computing and telecommunications industry sector. The sector serves the information processing, storage and networking demand of the digital economy. 5, 10, 11, 75, 76, 82, 90, 93, 95, 106, 111, 112, 118, 124, 125, 127, 130–132, 135, 137, 142, 143, 148, 149, 155, 157, 169, 188, 190, 193, 196, 197, 199, 200, 202

implementation

In the context of standardization, the term implementation refers to a product that is compliant with the specification of a standard. 97

intellectual property

Intellectual property (IP) is a term that describes intangible creations of the human intellect that can be controlled by an owning entity, like artistic works, inventions and designs. IP is made a tradeable good through the application of IPR. 76–78, 81, 82, 86, 87, 89–92, 103, 135, 151, 152, 155, 165, 176, 178, 193, 222, 233

intellectual property rights

Intellectual property rights (IPR) are an abstract concept that includes copyright, designs, patents, trademarks and other rights that are associated with IP. In most cases, explicit permission by the rights holder (a license) is required to use an IPR protected work.

IPR facilitate the trading of rights to use intangible assets. They are utilized to control permission to use the work through licensing and other relationships. 4, 6, 9, 13, 75, 76, 79, 82, 111, 117–119, 127–131, 135, 136, 139, 144, 176, 183, 188, 190, 191, 193, 198–200, 222, 223, 226, 233

International Organization for Standardization

The International Organization for Standardization is an international, independent, non-governmental standard setting body that consists of delegates from national standards bodies, with headquarters in Geneva, Switzerland. 81, 83, 94, 103, 104, 121, 128, 130, 136, 143, 189, 190, 224, 233

interoperability

Interoperability describes "the ability to transfer and render useful data and other information across systems [...], applications, or components"[57] 93, 97, 106, 108, 109, 118, 132, 143, 154, 162, 169, 188, 191

invention-implementation time gap

The time between the inception of an invention and when it is finally adopted in the market is called the invention-implementation time gap. IPR frameworks incentivize entrepreneurial activities by establishing temporary monopolies that enable rights holders to recover the necessary investments during the invention-implementation time gap in the subsequent monetization period. While traditional IPR frameworks assume the time gap to be in the order of magnitude of months to years, the gap does not exist in the case of openly developed FOSS technologies. 200

joint stewardship

The contributors who currently develop and maintain a FOSS project are typically a subset of all of the copyright holders. The current group of contributors assumes *joint stewardship* over the technical development of the project and the management of the community. The currently active stakeholders steer the project on behalf of themselves and the previous contributors who are likely copyright holders on the code, but who no longer participate in the project. 194, 195, 201

KDE

KDE is a recursive acronym that refers to the "K Desktop Environment", a graphical computing environment for FOSS operating systems. It is produced by the KDE community[8], a global, volunteer driven and decentralized FOSS community. 25, 29, 35, 43, 49–52, 55–57, 64, 65, 67, 185, 217

KDE e.V.

KDE e.V.[9] is a non-profit organization registered in Berlin, Germany that represents the KDE Community in legal and financial matters. 25, 26, 29, 50–52, 54, 56, 57, 64, 65, 67, 70, 102, 228

Legal Network

The FSFE *Legal Network*, initially called the European Legal Network, is a "neutral, non-partisan, group of experts in different fields involved in Free Software legal issues. Currently the Legal Network has over 400 participants from different legal systems, academic backgrounds and affiliations".[10] v, 8, 44, 145

Linux distribution

A Linux distribution is a software collection based on Linux that aggregates the work of the wider open source community into a complete operating system. Linux distributions are an integral part of the upstream/downstream model because they make FOSS accessible to specific applications like end-user desktops or embedded systems. 100, 123

Linux Foundation

The Linux Foundation is a not-for-profit industry association (US 501(c)(6)) dedicated to building sustainable ecosystems around open source projects to accelerate technology development and commercial adoption. 106, 117, 124, 135, 145, 220, 228

meritocracy

In the context of FOSS, the term meritocracy is used to describe a system where contributors gain reputation in a community solely based on the value of the contributions they make. 28, 225

national standards body

Countries nominate one SSO that represents them in ISO. These appointed SSOs are referred to as national standards bodies. They often have a special privileged relationship with their country. For example, a treaty between the federal republic of Germany and DIN declares DIN to be the national standards body of Germany, and in

[8] `https://www.kde.org` (accessed 03/02/2024)
[9] `https://ev.kde.org/` (accessed 03/02/2024)
[10] `https://fsfe.org/activities/ln/` (accessed 03/02/2024)

turn obligates DIN to the public benefit, among other terms. 94, 119–121, 130, 134, 136, 142, 143, 189, 223

open source

The term "open source" is generally used in this paper and many other contexts as a synonym to FOSS. It originally describes a campaign to promote free software to business.[113] 1, 3, 4, 27, 96, 112, 118, 119, 185, 186, 188, 197, 227, *see* FOSS

open source culture

A synonym for the open source way. 100, 123

Open Source Definition

The Open Source Definition formulates the terms software must comply with to be considered FOSS.[11] It is maintained by the Open Source Initiative. By way of the Open Source Definition, especially it's unanimous acceptance, the term "open source" gained a precise meaning across the wider open source community, and is therefore a term of art and part of open source culture. 1, 7, 89, 100, 153, 193, 197, 219–221, 225, *see* FOSS

Open Source Initiative

The Open Source Initiative[12] was founded in 1998 by Eric Raymond and Bruce Perens. It is dedicated to the promotion of open source software. The term "open source" was coined by the initiative's founders. It created the initial Open Source Definition. Today the Open Source Initiative is the steward of the Open Source Definition (OSD) and the community-recognized body for reviewing and approving licenses as OSD-conformant. It publishes the list of all approved open source licenses.[13] The Open Source Initiative is a California public benefit corporation, with 501(c)3 tax-exempt status. 27, 50, 100, 145, 153, 174, 197, 221, 225

open source way

The colloquial term "the open source way" describes the mutual understanding of software freedom, values and principles applied by the wider open source community, especially the norms of non-negotiable FOSS licenses, open governance and meritocracy. The common saying "There is more than one open source way" highlights that communities operate differently based on these shared values. 100, 123, 225

[11] `https://opensource.org/osd` (accessed 03/02/2024)
[12] `https://opensource.org` (accessed 03/02/2024)
[13] `https://opensource.org/license` (accessed 03/02/2024)

open standard

The European Interoperability Framework requires for open standards to give all stakeholders the opportunity to contribute to the development of the specification, the availability of the specification to everybody to study, and for the relevant IPR to be licensed "on FRAND terms, in a way that allows implementation in both proprietary and open source software, and preferably on a royalty-free basis".[46] 102, 134

open-source software steward

An open-source software steward as defined by the CRA is a legal person, other than a manufacturer, which systematically and on a sustained basis supports FOSS products. Most foundations will be open-source software stewards. Stewards have fewer obligations under the CRA than manufacturers, however they are still responsible for, for example, implementing cybersecurity policies and processes. 3, 185

partial consensus

Partial consensus refers to the social norms required to be followed for contributors to participate in a community. Most commonly, contributors need to accept the governance norms, the technical vision and the adopted license of the community. Complete alignment on political, economic and societal goals of participation is not necessary. Achieving it would possibly be in conflict with the goals of openness and inclusion. 186

public good

A public good is a good that is both non-excludable and non-rivalrous. The application of a FOSS license makes software source code a public good. 18, 30, 93, 99, 101, 142, 143, 221

release-early-release-often

Release-early-release-often describes an approach to software development where software is released early in the development process to start a feedback cycle, as opposed to after functional development is completed. The concept was popularized in the wider open source community. 7, 190

rivalrous

A good is rivalrous if its consumption by one consumer prevents or inhibits simultaneous consumption by other consumers. Most physical goods are rivalrous, while most information goods are non-rivalrous. 18

small and medium-sized enterprises

Small and medium-sized enterprises are defined in the EU as "enterprises which employ fewer than 250 persons and which have an annual turnover not exceeding EUR 50 million, and/or an annual balance sheet total not exceeding EUR 43 million."[49] 178, 196, 235

software freedom

"Software freedom" is what distributing a work under a FOSS license affords the user. While the terms free software and open source are used mostly synonymously today, "software freedom" is a political goal elements of the free software movement aim for. It can be circumscribed as the absence of coercion to use proprietary software.[140] Organisations like the Software Freedom Conservancy[14] and the Software Freedom Law Center[15] work to advance software freedom. 6, 7, 27, 34, 39, 43, 70, 118, 137, 139, 140, 183, 193, 198, 225

specification

In the context of standardization, the term specification refers to a set of documents that describe the requirements to be satisfied by a technical standard. 97

standardization

Standardization describes an activity of establishing, with regard to actual or potential problems, provision for common and repeated use, aimed at the achievement of the optimum degree of order in a given context (EN 45020:2006). 222, 227

standardization instrument

A standardization instrument is a mechanism applied by stakeholder that causes a standardizing effect. Examples for standardization instruments are recognized SSO, normalized customs and practices enforced by tradition, codes of behavior that are prevalent in some industry sectors, especially trade, but also industrial consortia, professional charters, or FOSS governance. 94, 96, 104, 105, 108, 138, 143, 144, 190, 227

standardizing effect

A standardizing effect is observable in the adoption of a common technical solution that results from the application of a standardization instrument. 93, 95, 98, 104, 105, 137, 143, 144, 190, 227

[14] https://sfconservancy.org/ (accessed 03/02/2024)
[15] https://www.softwarefreedom.org/ (accessed 03/02/2024)

standards development organization

A standards development organization is an entity whose primary activities are developing, coordinating, promulgating, revising, amending, reissuing, interpreting or otherwise maintaining standards that address the interests of a wide base of users. 17, 77–87, 89–92, 188–192, 219, 220, 234, 235

standards essential patent

A standards essential patent claims an invention that must be used to comply with a technical standard. SSO establish policies that regulate towards their participants how standard essential patents are expected to be licensed. 4, 77, 79, 81, 86, 87, 89, 90, 102, 117, 118, 128, 131, 136, 139, 142, 168, 190, 191, 199, 234

technical steering committee

A technical steering committee coordinates development and guides the community towards a technical roadmap. It is commonly set up to be independent from the project's governing board to ensure that development decisions are made based on their technical merit, avoiding potential conflicts with the business interest of funding members of the project. 222, 235

umbrella organization

An umbrella organization hosts individual projects within a single administrative structure. The Eclipse Foundation, the Linux Foundation and KDE e.V. are examples for umbrella organizations for FOSS projects 84, 91, 92, 106, 117, 121, 124, 132, 135, 141, 143, 144, 185, 192, 217, 220

upstream/downstream model

The analogy of the upstream/downstream model uses the mental image of a large river that collects the water from many smaller and smaller tributaries (the communities) and delivers it to the ocean (the users). Key tenets of the upstream/downstream model are the non-negotiability of the free software licensing terms and community governance norms. 93, 101, 107, 118, 132, 141, 143, 164, 194, 224, 228, 229

upstream/downstream network

Upstream/downstream network is another term for upstream/downstream model. 164, 165, 175, 176, 178, 183, 190, 191, 194, *see* upstream/downstream model

voluntary participation

Contributions to FOSS communities are made based on voluntary participation, both for individuals and for corporate contributors (organisations). Individuals are free to decide for themselves whether or not and what to contribute. Organisations as well decide what to invest and where and direct their staff accordingly. Community governance needs to account for voluntary participation as usually participants need to be convinced to make contributions, as opposed to being under obligations to do so. 18, 27–30, 41, 71, 72

volunteer contributor

Volunteer contributors participate as individuals following their own personal motivation, as opposed to corporate contributors that follow business rationale. Volunteers are free to decide on their own how much of their time and resources to invest in which activities. 2, 18, 25, 26, 51, 55, 61, 63, 64, 73, 219, 222

wider open source community

The phrase *wider open source community* is commonly used to describe the individuals, smaller and larger communities, businesses, umbrella organizations and other entities collaborating on developing the commons of FOSS in the global upstream/downstream model. 6, 10, 18, 28, 42, 43, 48, 70, 82, 93–95, 100, 106, 115, 116, 118–122, 124, 127, 131, 134, 138, 140–145, 147, 183–188, 192, 193, 195–197, 199, 201, 202, 217, 224–226

Wikimedia Deutschland e.V.

Wikimedia Deutschland e.V. is a non-profit organization registered in Berlin, Germany that represents the German language Wikipedia community. 58–65, 70, 71

Wikimedia Foundation

The Wikimedia Foundation is a US-based non-profit organisation that represents Wikipedia globally. 59, 60, 220

Wikipedia

Wikipedia is an online encyclopedia that aims at making the knowledge of the world available to everybody. 43, 58–62, 65, 70, 71

World Wide Web Consortium

The World Wide Web Consortium is an international community that develops open standards to ensure the long-term growth of the Web. 79–81, 87, 103, 106, 235

Acronyms

AI
: artificial intelligence 197

API
: application programming interface 169

ASL-2.0
: Apache Licene 2.0 87, 89

BSD
: BSD License 78, 88

BSD-2-clause
: BSD 2-Clause "Simplified" License 87, 88

CLA
: contributor license agreement 118, 166, 170, 179, 185, 195, 218

CRA
: Cyber Resilience Act 3, 4, 226

CWA
: CEN Workshop Agreement 129

DIN
: Deutsches Institut für Normung e.V 112, 117, 130, 145

EC
: European Commission 82, 118, 121

ECMA
: Ecma 83, 190

GPL

GNU General Public License 9, 77, 87, 88, 97, 125, 126

GPL-2.0

GNU General Public License v2.0 80, 87, 90, 126

GPL-3.0

GNU General Public License v3.0 89, 90, 125, 126

IEC

International Electrotechnical Commission 81, 94, 104, 190

IEEE-SA

Institute of Electrical and Electronics Engineers Standards Association 112, 116, 145

IETF

Internet Engineering Task Force 80, 103

IP

intellectual property 76–78, 81, 82, 86, 87, 89–92, 103, 135, 151, 152, 155, 165, 176, 178, 193, 222

IPR

intellectual property rights 4, 6, 9, 13, 75, 76, 79, 82, 111, 117–119, 127–131, 135, 136, 139, 144, 176, 183, 188, 190, 191, 193, 198–200, 222, 223, 226

ISO

International Organization for Standardization 9, 81, 83, 94, 103, 104, 121, 128, 130, 136, 143, 189, 190, 224

ITU

International Telecommunication Union 81, 94, 110

LGPL-2.1

GNU Lesser General Public License v2.1 88, 90

LGPL-3.0

GNU Lesser General Public License v3.0 88, 90

MIT

MIT License 78, 87, 88

OASIS

Organization for the Advancement of Structured Information Standards 77, 79–81, 87, 103, 112, 145

OIN

Open Invention Network 90, 122, 191

OpenDocument

Open Document Format for Office Applications 103, 106, 132

OSS

open source software *see* FOSS

PMPC

public money - public code 119, 142, 201

R&D

research and development 5, 91, 92, 107, 140, 156, 157, 169–172, 176, 181, 192, 198, 200

RAND

reasonable and non-discriminatory *see* FRAND

RDFa

Resource Description Framework in Attributes 80

RDFa-1.0

Resource Description Framework in Attributes 1.0 80

RF

royalty-free 2, 77–81, 84, 87, 89

SDG

Sustainable Development Goal 11

SDO

standards development organization 6, 17, 77–87, 89–92, 188–192, 219, 220

SEP

standards essential patent 4, 77, 79, 81, 86, 87, 89, 90, 102, 117, 118, 128, 131, 136, 139, 142, 168, 190, 191, 199

SFLC

Software Freedom Law Center 125

SME

small and medium-sized enterprises 95, 178, 196

SSO

standards setting organization 6, 93–99, 102–105, 109–121, 127–145, 224, 227, 228, *see* SDO

SWOT

strengths, weaknesses, opportunities and threats 96, 112–115, 188

TSC

technical steering committee 184, 222

UN

United Nations 11

VC

venture capital 173

W3C

World Wide Web Consortium 77, 79–81, 87, 103, 106