THE COMMUNICATION OF ECONOMIC RATIONALITY

IN VOLUNTARY CORPORATIONS

A Dissertation Presented

by

NIMROD SHAVIT

Submitted to the Graduate School of the
University of Massachusetts Amherst in partial fulfillment
of the requirements for the degree of

DOCTOR OF PHILOSOPHY

FEBRUARY 2021

Communication

THE COMMUNICATION OF ECONOMIC RATIONALITY

IN VOLUNTARY CORPORATIONS

A Dissertation Presented

By

NIMROD SHAVIT

Approved as to style and content by:

_____
Donal Carbaugh, Chair

_____
Benjamin Bailey, Member

_____
Jonathan Rosa, Member

_____
Erica Scharrer, Department Head
Department of Communication

DEDICATION

To Steven Weber whose original work on open source software inspired the writing of this dissertation.

ACKNOWLEDGMENTS

I thank my wife and parents, my adviser and committee members, and my friend Tyler Boudreau, for making this work possible.

ABSTRACT

THE COMMUNICATION OF ECONOMIC RATIONALITY

IN VOLUNTARY CORPORATIONS

FEBRUARY 2021

NIMROD SHAVIT, B.A., UNIVERSITY OF HAIFA

M.A., UNIVERSITY OF HAIFA

Ph.D., UNIVERSITY OF MASSACHUSETTS AMHERST

Directed by: Professor Donal Carbaugh

This Cultural Discourse Analysis (CuDA) examines how practices of open source software production articulate the causal relationship between economic rationality and social organization. The empirical question asks how is it that programmers who choose to collaborate under conditions of time scarcity and lack of command ability manage to create a durable organization of production. The examination of actual practices of participants shows that free and open source software production is driven by a rational-instrumental desire for utility maximization. While individual self-interest depends on local communication practices for its articulation, it remains prior to both culture and communication. The study therefore concludes that there are constant human nature factors which are not, themselves, socioculturally determined, and that the acceptance of such factors is necessary for the development of a theory of human agency within communication studies.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

THE SOCIAL UNIT OF AN EKNIGHT

A group of Israeli volunteer software developers in Tel Aviv, known as The Public Knowledge Workshop (PKW), have independently undertaken to build internet websites through which detailed information pertaining to government activities and officials may be made more transparent and accessible to the Israeli public. PKW mission is to provide other civic actors such as investigative journalists with specialized tools for the enforcement of public accountability.[1] While there are older and more established open government data organizations performing similar tasks, PKW stands out among them for the local means by which its members achieve their mission.[2]

---

[1] In the rhetorical terminology of Lloyd Bitzer (1968) and Carolyn Miller (1984), PKW creation and operation of civic websites can be considered as a recurrent visual-rhetorical response to typified exigences of government opacity, inaccessibility and corruption.

[2] This is not the place for a comprehensive review of the open government data movement and the research about it, as the present study is concerned primarily with the relationship between economic rationality and social organization. Suffice is to say that the open government data movement started in post WWII U.S., with the rise of what historian of communication Michael Schudson (2015) proposed as *the cultural right to know* — i.e., a historically unprecedented political climate in the Anglophone West where the demand for "transparency" of government institutions, social organizations, commercial firms and the like reflects a taken for granted "right." The most well-known product of this movement is the Freedom of Information Act that provides American citizens with a principal access to archives and databases of public institutions. More recently, this movement experienced a revival with the introduction of computer technologies that allow activists, journalists and ordinary citizens to organize and analyze datasets whose study was practically impossible in the past. The American

PKW develops its civic websites through small project teams known by the local term "xashmabirim" or "electroknights" [from the Hebrew words "xashmal" (electricity) and "abir" (knight)], which is sometimes abbreviated to the English term "eKnight/s." Within PKW volunteer setting, there may be multiple project teams operating simultaneously on completely separate civic websites, each of which is referred to as its own eKnight.[3]

Sunlight Foundation and the British Open Data are examples of this trend. Additionally, recent years have saw a rise of closely related organizations whose goal is to provide accessibility to public and governmental services by showing officials how computer technologies may help to increase the efficiency of bureaucratic procedures. The organization Code for America provides an example of such an initiative. While sharing important similarities with these other groups, PKW developed its own organizational mechanism for civic software production. For further reading on open government data, see the excellent report of programmer and activist Joshua Tauberer at opengovdata.io and the pioneering collection of academic essays on open government data organizations edited for New Media & Society by Jeremy Hunsinger and Andrew Schrock (2016). For a critical evaluation of open government data and its key notion of openness see the brilliant analysis of Nathaniel Tkacz (2012).

[3] The terms eKnight (in singular) and eKnights (in plural) are borrowed from the science fiction book The Cyberiad (1974) by Polish writer Stanisław Lem, which tells, among other things, about machinic knights that function as public servants and guardians in an imaginary universe inhabited by artificially intelligent robots. Much like the intelligent robots in the story, PKW participants envision themselves as technology craftsmen or "code artisans" (xarashei cod) who build automatic eKnights whose function is to serve and protect the Israeli polity. This ideology of the common good is premised on two basic assumptions. First, insofar as the eKnights are aimed to serve all the citizens in the polity, they must remain free from ownership and control by public and private agencies. Secondly, because different citizens in the polity have particular needs that cannot be known in advance and may change over time, the eKnights must remain open to reconfiguration so as to accommodate those various and changing needs. Importantly, not all of PKW participants use this term when they speak about their projects, and at least some of them have reservations about the civic ideology it reflects. In using the words eKnight and eKnights I therefore do not attempt to suggest a consensual agreement among all the participants in PKW volunteer setting. Rather, my purpose is to draw attention to the features that make the projects that emerge in this particular volunteer setting distinct from other kinds of software

The volunteers in these eKnights are associated administratively to PKW, but function mostly within the confines of their individual teams with no official affiliation to the organization at large. Those participants will typically gather together on a weekly basis for "development meetings" held in community spaces where individual programmers come to work on their respective projects. These development meetings run for roughly five hours and are held after the regular business day is over so that participants may come after they have finished work.

PKW eKnights are the social units in which this study is anchored. Analytically, these units can be best defined as *voluntary corporations* in the original sense of the term corporation, i.e., "a group constituted for material operations, some of which involve social engagements and relationships" (Bird-David 1997:471-472).[4]

In PKW volunteer setting, eKnights qua voluntary corporations have three features that make them distinct from other, perhaps more conventional, forms of volunteering. First, the group's eKnights are organized and managed by the volunteers themselves, not by the organization's administrators. PKW as an official Voluntary Association does not maintain any authority or

---

projects, and especially the for-profit projects of the commercial firm.

[4] In this economic anthropological framework, corporations are contrasted with households qua "socially constituted groups, which in maintaining themselves engage in material actions" (Bird-David 1997:471). For the basic orientation of this approach, which shares much with cultural communication frameworks within communication studies, see Gudeman (1986) and Gudeman and Rivera (1990).

control over the projects or their participants. Rather, the group's administration is locally conceived as a "startup incubator" whose role is to provide the different eKnights with legal, logistical and public relations services. Secondly, each of PKW eKnights remains independent from the others, both socially and in the execution of programming tasks. In other words, there is no particular sense of community among the project teams even though their development meetings are co-located. Finally, PKW software products are created almost exclusively within the designated times and spaces of development meetings. PKW eKnights are therefore the primary mode of production, rather than a secondary mode in which volunteers might come and contribute to the paid labor of other civic or social actors.

The Problem of Voluntary Corporation

At first glance, the distinct features of PKW eKnights may appear to represent little more than an interesting work arrangement. However, these same features are quite counterintuitive in terms of the social conditions they reflect, under which PKW participants operate, and whose radical nature can be best described through the lens of traditional microeconomics. In conducting this description, I seek to bring microeconomic theory to the communication field of Cultural Discourse Analysis (CuDA). The relationship between the two seemingly disparate approaches is discussed and clarified below.

As is widely agreed, microeconomic thought is premised on the assumption that human beings have limited means or resources by which they can satisfy only some of their unlimited wants. To cope with this condition of scarcity, people must rely on their capacity for rational action and thought. They need to plan their actions by analyzing the relationship between that which they want and that which they must give up in order to get it. To the extent that people are rational, they are expected to pursue the fulfillment of a given want only if their *utility* or the satisfaction they may gain from this fulfillment is greater than their *disutility* or the dissatisfaction they may experience when using the limited resource/s by whose expenditure the fulfillment of that satisfaction is conditioned. Hence the principle of rational action/choice that provides, for the economist, the explanatory basis of all social behavior.

From this theoretical standpoint, the lack of remuneration for participants' work and contributions in PKW volunteer setting is not the central puzzle needing explanation. An individual programmer who chooses to develop a piece of software without monetary compensation may very well gain some other utility whose value is greater than the sum of his production costs. Indeed, I found as part of this research that Israeli software developers choose to volunteer in PKW eKnights for a variety of mundane reasons that include but are not limited to the provision of collective goods that they value, the solution of civic problems that bother them personally, the satisfaction of their curiosity

about various technical and social issues, the reclamation of civic agency in what they sometime perceive as an apolitical social environment, the acquisition of new technical skills, and the creation of professional alliances that could help them promote their careers in the high-tech industry.[5]

---

[5] These individual motivations are rather trivial and can hardly form the topic of a dissertation project. Additionally, the utilitarian reasoning of individual participants in voluntary corporations and of volunteers more generally has been studied extensively. Contemporary microeconomists largely agree that volunteers are "impure altruists" with both individual and social interests (Hustinx et al. 2010:416). The individual interests are: (i) the enhancement of one's human capital through the acquisition of skills that one cannot easily acquire without monetary costs (e.g., Freeman 1997); and (ii) the emotional experience of group solidarity that can only be achieved in non-market settings (e.g., Andreoni 1990; Rose-Ackerman 1996). Closely related to this second individual interest is the social interest to advance a cause or an ideology to which one adheres together with other believers (e.g., Duncan 1999).

On a parallel line of research, political economist Steven Weber (2004) has attempted to explain the economic interest of individual participants in voluntary corporations that create free and open source software such as PKW eKnights. Weber's explanation starts from the observation that open source software products are non-excludable and non-rivalrous goods. Non-excludable because they are not owned by anyone (at least not in the traditional sense), and non-rivalrous because the usage of one person does not deprive anyone else from using the same piece of software. In fact, OSS products such as the Linux Operating System are an extreme version of non-rivalrous goods due to one's ability to replicate them infinitely at an almost zero cost. Based on this observation, Weber (2004:154) proposed that the use value of a given OSS to an individual user increases as more individuals choose to use the same software on their machines. There are two reasons for this. First, just as it is more valuable for one person to have a fax machine if many other people also have fax machines, "as more computers in the world run a particular operating system or application it becomes easier to communicate and share files across those computers." In this view, any copy of a given open source software "becomes slightly more valuable to existing users as each new user enters the picture." The second reason has to do with maintenance and debugging. In Weber's words (2004:154):

> The more users (and the more different kinds of users) actively engage in using a piece of software, the more likely that any particular bug will surface in someone's experience. And once a bug is identified, it becomes possible to fix it, improving the software at a faster rate. This is hugely important to the

That which unites the participants in the typical eKnight is therefore not a common utility; rather, it is the common interest of rational actors to gain their different utilities — whatever these may be — at minimal production costs. The novel set of communication practices that constitute and mediate this

---

economics of software users, because customization, debugging, and maintenance usually accounts for at least half (and sometimes considerably more) of the total cost of ownership of enterprise software.

In this calculation, open source software is not simply a non-rivalrous good that can tolerate free riding without reducing the stock of the good for its makers. Instead, it is an "anti-rivalrous good" in the sense that "the system as a whole positively benefits from free riders" (ibid). Insofar as a free rider who chooses to use a particular piece of software increases the use value of that software for everyone else who use it, any incidental contribution by this individual (e.g., a self-serving report on a bug) has the potential to increase that use value even more. In this view, if a small group of programmers seek to create a new OSS product, it would be more beneficial for them to distribute that software to a large audience of consumers with the assumption that at least some users will develop a dependency on it and might also offer occasional contributions to it in the form of bug reports or bug fixes.

To be sure, an important limitation of this explanation is its inability to account for the choice of participants to completely give up remuneration. Assuming that scenarios in which the creation of use value radically contradicts the accumulation of wealth are rare, it is unclear why software developers will not try to make at least some monetary profit from their labors. For this reason, the utility of increasing the use value of one's software product cannot stand on its own as a core motive. It can, however, be added to the individual and social utilities specified above and to any other material utility for that matter. Weber's microeconomic terms together with the banality of self-interest among PKW participants as well as the basic fact that there is no readymade market for such civic products in Israel suggest that an elaborated analysis of PKW discourse of motives would be a waste of time. A much more interesting way to approach this question would be to follow Hopper (1993) in contrasting between microeconomic and rhetorical theories of human motivation. For example, one could ask if there is any analytically rigorous way to prove that a given motive is a socially constructed fact rather than a want or desire that emerges in one's mind prior to and outside of any sociocultural context. Given that the present study was not designed for this sort of examination, I leave it for a future work.

common economic interest warrants serious study which this dissertation provides.

Time Scarcity and Lack of Command Ability

Production costs in PKW volunteer setting are estimated in units of free time — i.e., a scarce resource whose factuality derives from the structural opposition between work and leisure (or play) in modern industrial society (Turner 1982). The volunteer work of modern actors is by definition an investment of leisure time and, as such, is secondary to vocational and familial subsistence obligations. This is more so when one takes into consideration that many PKW volunteers are family men in their thirties with small children and demanding jobs in the high-tech industry. These individuals can only dedicate several weekly hours to their projects and must therefore prioritize the activity of software production over any other social activity in the development meetings if they are ever to complete their tasks and gain their individual utilities.

While it is no surprise that the volunteers in PKW eKnights are driven by a common economic interest, it is not as clear how they manage to work together and cooperate in an efficient and effective fashion. As is apparent from the paradigm of the commercial firm, efficient organization of production requires a strong institutional order. In modern capitalist society, firm workers rent their labor capacities to an employer for a set amount of time in exchange for a wage. If they excel in their

jobs (i.e., increase the profits or market values of their companies), they may be given a raise or promotion. If they do not follow the rules, they will be subjected to material and social costs and may eventually be fired. These institutional capacities for punishment and reward loom in the background as deterrents of antisocial behavior among participants in any commercial organization of production.

Participants in voluntary corporations such as PKW eKnights cannot reproduce the hierarchical structures and institutional orders by which commercial firms organize their production processes because no participant in any such group has a social mandate to tell somebody else what to do. There are at least two reasons for this. First, in contrast with workers of commercial firms, participants in voluntary corporations do not depend on their initiatives for their material subsistence.[6] One can either volunteer to an eKnight or not with little consequences for his livelihood and career. The second, more essential reason, derives from the fact that participants in voluntary corporations have almost no ability to exercise power, provision rewards, enforce sanctions, or force people to actions. If H tells S to do

[6] From a strict microeconomic perspective, it can be said that participants who exit their projects in such a premature manner do suffer costs; however, one must keep in mind that these costs are relatively low. The greatest risk to members of modern voluntary corporations is a loss of some leisure time and a potential failure to satisfy less essential or surplus desires that reach beyond basic material requirements (e.g., food, shelter, etc.), a mode of prioritization that appears to correspond well with Maslow's (1943) famous "hierarchy of needs."

something and S is not doing this, H cannot sanction or punish S nor can he reward him in any socially significant way. Hence the fundamental condition that I propose to account for the set of practices constitutive of PKW organization of production, in the following, as *lack of command ability*.

The condition of lack of command ability results from the nature of political power in modern industrial society. Here, power is not only entwined with the rule of law (i.e., legal systems and policies of institutional workplaces) but also with liberal and democratic ideals of freedom and equality. Truly endogenous voluntary corporations — especially ones in which strangers with weak social ties gather together to perform a task — presuppose a symmetrical relation among all parties involved and are thus egalitarian by default.[7] In this situation, there can be no institutional order that forces someone to do something against his or her will.

To the extent that members of voluntary corporations know that they have no social mandate to tell others what to do, they are expected to refrain from doing so. This normative status quo is eloquently described by Joseph who initiated two very similar and successful projects in the American group Code for Boston:

(1) Interview (6/11/2015)

[7] This seems to be true to most modern societies in Durkheim's ([1893]1984) original sense. Within CuDA, such societies are commonly associated with a personal style of communication that Gerry Philipsen (1987) contrasts with the collective styles by which members of positional and traditional societies communicate with each other. For an extensive research on one such style in contemporary American society, see Carbaugh (1988).

1. [In a company] you are being told what to do, you have a hierarchy, you have a boss, you have a target to meet, you have a goal. [In the voluntary software projects] all of that doesn't exist […] Instead of being asked what to do, you do things because you want to, when you want to, if you want to.
2. So you cannot tell anybody […] what to do. I wouldn't think of going to one of my peers […] and say hey by the way blah blah blah you need to finish this screen because we need to ship this next week […] I will never say that. And that changes the dynamic a lot because you […] can push as much or as little as you want without repercussions or without commitment of any kind. The only commitment is that you [do what you do].
3. Does that mean that I am committed to keeping the contributions of the project going? No. Does that mean that I need to ship this in two weeks? No. Can this project die tomorrow? Absolutely it can die tomorrow […] If you really care about why you fill that project then you also care that it's still alive and that it's well and that it keeps going forward.

Principal Guiding Question

As Joseph indicates in the above excerpt, members of voluntary corporations such as PKW eKnights are faced with a concrete, practical problem. On the one hand, they share a basic economic interest of creating an efficient organization of production that would allow them to make the most out of the scarce leisure time they are willing to invest in their labors. But on the other, these same individuals must accept the absence of any authoritative structure in their project teams. The problem for PKW participants is therefore how to scale the ability to work together and cooperate in a temporally limited and spatially transient setting that presupposes the absence of authoritative and hierarchical structures as a condition for its possibility.[8]

[8] Here, it is important to note that PKW eKnights have existed for about ten years, and that some of these projects involved the creation of relatively complex software systems. In other words, even though the

This practical problem can be formulated in terms of the two analytically distinguishable components of any organization of production that I propose as *technosocial system* and *system of governance* (cf., Weber 2004:1). A technosocial system involves the production resources at use (including the corporation's workforce), and centers on the division of labor and the procedures by which workers coordinate their separate tasks and maintain standards of quality. A system of governance adds to this a set of institutional mechanisms that function to enforce the firm's regime of work, resolve conflicts between its members, and distribute social and economic value among these individuals (e.g., social statuses and shares of produce).

The analytical question then becomes, how do PKW participants manage to create an efficient technosocial system and an effective system of governance under extreme conditions of time scarcity and lack of command ability? From the standpoint of communication studies, this question leads to a data-based

---

group's eKnights are a marginal phenomenon in Israel, it is clear that the organization of production common to them has remained stable over the years. The very existence of the projects therefore arises in contradiction to one's commonsensical expectation that voluntary corporations will fail to create a sustainable organization of production. The endogenous methods by which participants in a single eKnight manage to overcome the essential problem of social organization can thus be conceptually generalized as a voluntary mode of production that arises in modern industrial society relative to the extant capitalist mode of production. The force of conceptual generalizations of this sort has been repeatedly demonstrated in the discipline of communication studies by scholars working within the subfield of Conversation Analysis (e.g., Schegloff 1987). In Rawls's (2004:136) observation, such studies reflect and confirm the Durkheimnian position that "if the detail to support [one's] argument cannot be found in a close examination of a single case, then it does not matter how many cases are piled up, it will not be found."

inquiry into the actual practices by which real people constitute a social order of work and production outside the traditional boundaries of both market and non-market institutions.

Theoretical Import

By providing a concrete answer to the question posed above, this study contributes to social scientific theory in general, and more specifically, to theory building within Cultural Discourse Analysis (CuDA) as a subfield of communication studies. The general contribution to the social sciences is twofold. First, the study in hand provides a detailed examination of the causal relationship between the explanatory variable *economic rationality* and the response variable *social organization*. Secondly, the study shows that this causal relationship relies on the mediating variable *cultural discourse* (as defined in the method chapter 3) for its efficacy. In so doing, the study confirms the central hypothesis of CuDA that communication is a building block of society and should therefore be included in the development of any social theory.

The study's more specific contribution to CuDA concerns the relationship between the microeconomic notion of human agency as a constituent of social interaction, and the constructionist finding that individual identities are products of such interactions. The main theoretical finding here is that the self-interests of an individual rational actor could place constraints upon socially constitutive communication acts and events.

To unpack the significance of these cross-disciplinary and intra-disciplinary contributions, the following discussion first situates the study within the broad panorama of social theory. It then explains how the microeconomic principle of rational action/choice can be used to develop a theory of human agency capable of accounting for the emergence of specific sociocultural arrangements within CuDA.

Contribution to the Social Sciences

While the idea that social organization tracks economic rationality is not new, it has remained contested within social scientific research. Let us recall that the rise of the academic disciplines of sociology and anthropology in 19th century Europe was, to an extent, a reaction against the absolute dominance of neoclassical economics as a scientific perspective from which to account for acute issues of the time, issues that concerned the causes and results of the industrial revolution and the transition from traditional to modern society more generally.

Early sociologists such as Durkheim ([1893]1984) and Weber ([1930]2002) therefore took pains to show that the paradigm of an individual rational actor maximizing utility cannot lay a fully adequate foundation for a social theory of modernization.[9] In a parallel vein, when Bronisław Malinowski went to invent ethnography in his study of the Trobriand Islanders in 1915, his

[9] See discussion in Rawls (1992).

principal guiding question was the following: can Western economic tools be used for the study of primitive economies? The critical answer to this question formed the first book-length work of economic anthropology: Argonauts of the Western Pacific ([1922]2002). In this book, Malinowski argued that the academic neoclassical economics of his time was an ethnocentric perspective by which Europeans could understand themselves but not others. The theories of Western economists, he wrote (ibid:46), would do nothing to explain economic behavior in the Trobriand Islands:

> Another notion which must be exploded, once and for ever, is that of the Primitive Economic Man of some current economic textbooks […] prompted in all his actions by a rationalistic conception of self-interest, and achieving his aims directly and with the minimum of effort. Even one well established instance should show how preposterous is this assumption. The primitive Trobriander furnishes us with such an instance, contradicting this fallacious theory. In the first place […] work is not carried out on the principle of the least effort. On the contrary, much time and energy is spent on wholly unnecessary effort, that is, from a utilitarian point of view.

In response to such extra disciplinary critiques, and especially during the 1960's within The Chicago School of Economics, Gary Becker (1930–2014) and his associates began to demonstrate that the principle of rational action/choice was fit to explain "not only what is happening on the market and through monetary exchanges, but any kind of social behavior: learning, wedding, love, crime etc." (Caillé 2013:44). This demonstration of intellectual force was highly successful due to its analytical rigor and its ability to establish falsifiable claims.

This return to dominance of microeconomic theory led to a vehement debate between proponents of two competing schools within economic anthropology, which came to be known as "substantivism" (or institutionalist) and "formalism" (hence the infamous "Formalist-Substantivist Debate"). Members of the more veteran substantivist school followed the critical perspective of economist and historian Karl Polanyi in asking very similar questions to the one posed by Malinowski.[10] Advocates of the rising formalist school countered their assumptions and findings with great elegance and sophistication.[11]

---

[10] In his master work The Great Transformation ([1944] 2001), Polanyi observed that the rise of market capitalism in Europe presented humanity with an unprecedented attempt to "disembed" the economy from the institutional fabric of society. His central argument was that the consequences of this attempt are disastrous as individuals are now left to fend for themselves in an alienated and alienating competition. Later on, Polanyi (1957) developed his critical approach into an historical and anthropological framework where market capitalism is only one economic system that can be explained by the academic theories of neoclassical economics after which it is modeled. Polanyi's hypothesis was that the other two economic systems that he called "redistribution" and "reciprocity" cannot be explained by these theories because they are embedded in social, political and religious institutions and must therefore operate on entirely different logics of exchange. Such instituted or "substantive" economic systems can only be studied historically and ethnographically as they are found in non-Western tribal societies and in prior Western societies. The critical thrust of the substantivist program was therefore to show that modern Western capitalism is an historical phenomenon that has no universal validity. The central exponent of this approach in economic anthropology was Polanyi's student George Dalton (e.g., 1961; 1969). Marshall Sahlins's book Stone Age Economics (1972) can be considered as the final word in the formalist––substantivist debate. This, however, does not mean that Sahlins managed to resolve the dispute, as this book was published at a time when the argument between the warring camps lost its heat and gave way to a disciplinary interest in the effects of Western imperialism and colonialism on subjugated peasant societies. For a retrospective review of this more contemporary Marxist approach known as "articulation" see Gregory (2009).

[11] For a definitive statement of this approach, see Schneider (1974). Another central contribution to the debate was a series of essays by

Later on, cultural and economic anthropologists attempted to reformulate Malinowski's original question in purely epistemological terms. Inspired by Weber's cultural-interpretive approach to human motivation — and especially the application of this approach in his thesis of Protestant Work Ethic — these scholars went on to suggest that the dominant neoclassical economics in its entirety was largely, if not exclusively, a cultural and historical construct.[12]

---

social anthropologist Fredrick Barth (e.g., 1959; 1967). While not identifying himself as a formalist, Barth attempted to show that instituted approaches to the study of human economy lack a model of human agency. As a consequence, they cannot explain what motivates individuals to act the way they do and how people can promote their individual goals and interests while taking a hand in shaping the societies in which they live and operate. With this approach, which he called "transactionalism," Barth attempted to turn the tables on Polanyi's followers by showing how complex sociocultural systems of non-Western tribes can be explained in terms of the strategies by which individuals act to secure material and symbolic goods (e.g., food, sex, prestige, rewards in the afterlife, and so on). The critical thrust of this argument was that substantivist research produces endless descriptions and taxonomies that do not explain anything in particular and therefore have little value for theory building within the social sciences.

[12] This trend was heralded by Clifford Geertz's comparative studies of local market economies in Java and Bali (e.g., 1956; 1957; 1963; 1984). The general aim of this line of research was to test the Weberian hypothesis that any non-Western religious system that bears similarities to the one advocated by adherents of the Protestant Reformation (or the Calvinist movement) in 16th century Europe would contribute to the same processes of modernization that occurred in the West (Keyes 2002). While not questioning the foundations of Western economic theory, Geertz updated and reasserted the basic Weberian idea that human economic behavior is embedded in cultural systems of value and belief. The first to apply this idea in the critique of neoclassical economics and modern Western rationality more generally was Marshall Sahlins (1976; 1996). While taking an anthropological approach to the interpretation of cultural symbols and meanings, Sahlins did not complement his critical observations with ethnographic field research. It seems that such a disengagement from the detailed analysis of economic behavior among people who live and operate in Western capitalist societies is not unique to Sahlins's (or Geertz's) work, and is one of the main limitations of contemporary research

In a parallel vein, social and cultural theorists, starting with Malinowski and Durkheim's nephew and collaborator, Marcel Mauss, have attempted to develop models of non-market exchange or reciprocity. The aim of such models has been to show that human behavior is not only non-utilitarian (at least in the first instance), but also that the primary function of the economy is to create and affirm bonds of solidarity.

In response to such communitarian models of reciprocity, microeconomists who work within the experimental subfield of game theory have persuasively shown that there is no necessary contradiction between rational action and social solidarity

---

within the field of economic anthropology. Indeed, this bias can be observed in more recent studies of cultural economists who tend to (i) investigate the non-market economies of traditional tribes whose members could have never imagined the rise of Western capitalism; and (ii) juxtapose the detailed ideas and behaviors of those people against a rather abstract notion of self-interested market exchange so as to relativize and thereby question the taken-for-granted status of the latter on cultural and historical grounds. The limitation of this approach to epistemological critique was pointed out by Marcus and Fischer (1999:152-156) who argued that a more detailed comparison among multiple fieldsites within and between localized groups is in order (see also, Marcus 1995; 2011). Nevertheless, such monographs have offered indispensable contributions to the cross-cultural understanding of human economic behavior, and they have much in common with the most current research conducted within the field of cultural communication (see especially, Gudeman 1986; Gudeman and Rivera 1990; Povinelli 1993; Bird-David 1993; 1999). While not following the route proposed by Marcus and Fischer, I attempt to contribute to this research effort in three distinct ways. First, I refocus attention on the economic activities of modern actors. Second, I show the merit of analyzing such activities within a communication framework. Third, I show the advantages of keeping a distinction between the types of society in which such micro level activities take place.

insofar as the latter is understood to be a kind of collective good that can only be maximized through social cooperation.[13]

In a broader view, it is quite clear that the critiques of anthropologists and sociologists have had little effect on the expanding field of microeconomics, whose practitioners are now operating under an explicit agenda of "economic imperialism" (e.g., Hurtado 2005). The term "imperialism" in this formulation reflects the belief of economists that their theoretical models are superior to other forms of social scientific explanation, and should therefore be used to unify the disciplines under a single paradigm. The ultimate goal of economic imperialism is therefore not to exercise political dominance over the social sciences, but rather to achieve the kind of consensual agreement characteristic of the natural sciences, especially physics.[14]

---

[13] For a sophisticated application of rational action theory to the most solidary form of reciprocity that Sahlins (1972) called "generalized," see Takahashi (2000).

[14] The academic movement of economic imperialism has had a significant impact on the long-standing disciplines of sociology, political science and anthropology. In sociology, the work of Gary Becker and his associates was compatible with the already existing attempts of George Homans (1910–1989) and Peter Blau (1918–2002) to establish a theory of "social exchange" on rational utilitarian grounds. At the same time, it also influenced sociologists such as James Coleman (1926–1995) to dissolve the boundaries between the two disciplines, an attempt that reached its peak with the publication of Coleman's influential book Foundations of Social Theory (1990). In political-science, the self-interested model was promoted by scholars such as Robert Axelrod (1970; 1981; 2006) and has had a particularly strong influence on research conducted within the fields of international security and international relationships (e.g., Weber 1991). In anthropology, ethnographers such as Fredrik Barth (1928–2016), Harold Schneider (1925–1987) and Scott Cook used the self-interested model as means to challenge the hegemony of Karl Polanyi's institutionalist school of economic anthropology.

The impact of Economic Imperialism on communication studies, and especially the field of social interaction has been much more subtle, as social interactionists rarely identify themselves as proponents of

The present study does not seek to intervene directly into any of these debates. Rather, the analytical aim is simply to present evidence for a causal relationship between economic rationality and social organization through communication practice in the case of PKW eKnights. In accomplishing this aim, the study does not attempt to imply that economic rationality is the only or even the primary cause of all social organization. Rather, the intent is to initiate a productive debate among socioculturalists whose intellectual commitments and intuitions tell them that social organization can emerge largely beyond the self-interested actions and choices of individual rational actors. Accordingly, the larger purpose of this dissertation is not to antagonize colleagues, but rather to show them how communication practices can function as expressive vehicles for human economic rationality. This may lead to constructive reconsiderations of using microeconomic theory as an explanatory mode of inquiry within CuDA studies that focus on modern organizations and organizational practices. The word "modern"

---

the self-interested model, or associate their research with any of the literature mentioned above. Nevertheless, one can clearly find indications for such an influence when reviewing communication literature with this particular purpose in mind. While the scope and impact of this influence is hard to assess, some examples for its effects are the pervasive usage of the terms "costs," "benefits," "benefactors" and "beneficiaries" within the area of interactional sociolinguistics (e.g., Couper-Kuhlen 2014), the explicit attempt to use the self-interested model as a foundation for "politeness theory" within the area of socio-pragmatics (Brown and Levinson [1978]1987; Clark and Schunk 1980), and the application of Jeremy Bentham's "felicific calculus" and the idea that a rational cost-benefit calculation may account for the emergence of "interactional preferences" within the area of Conversation Analysis (Clayman and Heritage 2014; Clayman 2002:249).

holds the key in this formulation not only because it is hard if not impossible to find truly traditional societies in the contemporary world system (Marcus 1995), but also because many microeconomists and socioculturalists agree that modernity is the era of the homo oeconomicus par excellence.

Here, one needs to consider the ontological problem of rational action/choice: Is rational action/choice a property of the human condition as such, or is it an element of a cultural system common only to modern Western capitalist societies? While this problem may have little significance for the analysis of socially situated action, it does pose a serious dilemma when one wants to draw out the political implications of such actions. To avoid complication, this study temporarily accepts the two oppositional views with the assumption that they could coexist in some possible world. While the view that self-love is an essential property of the human condition may be true,[15] it does not follow that self-interested action can be understood or interpreted outside the historical circle of cultural values and beliefs in which it is embedded (cf., Rawls 1992).

By threading a thin line between microeconomic and sociocultural approaches to human communication, this dissertation seeks to say something interesting about each from the standpoint of the other. The claim is being made throughout

[15] One cannot ignore the obvious fact that this position was held by pre-modern philosophers and theologists in any region of the world. Here, one could think of Plato's Socrates and of the Jewish Talmudic sages, to name but a few.

that a careful consideration of communicative action may lead to an enlarged sense of how human communication constitutes organizational arrangements through the mediation of individual self-interest.

Contribution to Communication Studies

My argument that descriptive-interpretive approaches in the social sciences may benefit from the development of theoretical models that take individual rational action/choice as an explanatory principle is directed primarily toward researchers working within the field of Cultural Discourse Analysis (CuDA) where my own scholarship is situated.

Historically, CuDA finds its origin in Dell Hymes's linguistic anthropological research program that came to be known as the ethnography of communication (Hymes 1972a). In the early iterations of this research program, Hymes sought to construct a cross-cultural taxonomy of social components and units with the general aim of providing a cultural-rhetorical alternative to the then predominating formal-linguistic theory of generative grammar. Within the discipline of communication, Hymes's original program was revised and expanded by Gerry Philipsen and Donal Carbaugh. Whereas Philipsen took a Durkheimian approach to the study of communication as ritual (1987; 2002), Carbaugh took a more interpretive approach, influenced by the hermeneutic philosophy of Gadamer, to the study of communication practice as an element in a system of symbols and meanings (1989; 1995;

2007a). Presently, these theorists and their students are concerned with the question of sociocultural formation through communicative practice (Carbaugh 1995). Their basic underlying assumption is that communication practices or social activities of meaning making "play an essential role in constituting the ideational and material aspects of human existence and, hence, in bringing about particular ways of being-in-the-world" (Duranti 1997:4-5). To speak of communicative constitution is to observe that media such as words, images and sounds not only transmit information (Carey 1975) but also formulate exclusive connections among those they link, creating cultural forms of social life through activities of uptake and response at different scales of social history (Agha 2011:163).

While it should be taken for granted that cultural discourse analysts have developed highly specialized tools for the description and interpretation of social organization, and have thus made important contributions to our understanding of the constitutive role of communication in processes of sociocultural formation, this approach may be enhanced through further theorization of human agency and the relationship between cultural interpretation and economic explanation.

As we unpack these claims, it will be useful to consider some aspects of CuDA, in particular its focus on symbolic and structural components that define conditions and possibilities of human expression and identification in any social setting. CuDA typically bases its analyses on the assumption that individual

action is only partially determined by the sociocultural systems in which it is embedded. Under this assumption, a sociocultural system is considered to be *individually applied* as a condition for its possibility (Carbaugh 1995:280). Such individual application, according to Philipsen (1992:10), cannot (and should not) be theorized as socioculturally determined. In his words:

> To say that [communication] is structured is not to say that it is absolutely determined. It is patterned, but in ways that its creators can circumvent, challenge, and revise. Its rules are violated, new rules and meanings are created, and there in play is brought into structure just as structure is brought into play.

The difficulty here, especially relative to the questions being pursued in this dissertation, is the argument that human agency itself varies by sociocultural systems. I submit that if sociocultural systems do not predetermine individual actions, then (i) such actions must be purposive or goal-oriented at least in part; (ii) the thinking subjects who direct these actions must be rational even if in a rudimentary sense; and (iii) this rationality cannot be culturally specific, but rather a property of the human condition. The present study provides evidence in support of these three propositions.

A related potential limitation of the CuDA framework is its preference for thick descriptions of extant sociocultural systems. The original causes of such systems are, themselves, not generally the focal points of CuDA scholarship, leaving us with some important, but unanswered, questions to address. The study in hand proposes a microeconomic theory of human agency whose principles can be used to explain the emergence of sociocultural

and communication systems such as the one pertaining to PKW technosocial system and system of governance.

CODA

In the interest of readability, the introductory discussion of this dissertation is separated into three chapters. The present chapter has constructed the puzzle of voluntary corporation in relation to the theoretical regimes of microeconomics and CuDA. The following chapter presents and discusses the data on which the analysis in chapters 4 to 8 is based. Finally, chapter 3 outlines the procedure by which the analysis in these latter chapters is executed.

CHAPTER 2

FIELDSITES AND DATA

Introduction

The data on which the analysis presented in this dissertation is based was collected between the years 2014 and 2020 in the U.S. and Israel. As is the case with any CuDA research, the distinct features of the corpus at hand can be best understood in relation to the social contexts in which the data were generated. The following discussion therefore intertwines descriptions of these social contexts with methodological explanations of how the data were collected, documented and organized for the particular purposes of this study.

Before I begin, there are three compositional issues that need clarification. The first issue derives from the fact that most of my knowledge about PKW is limited to the years 2015 to 2016, the period during which the majority of the actual (as opposed to the virtual) data were generated (i.e., field-notes, field interviews and in-depth interviews). After that period, I kept collecting electronic data which can be considered as up to date. In the interest of readability, I ignore these temporal distinctions and speak of PKW primarily in the present tense.[16]

The second compositional issue is my default usage of the masculine pronouns "he," "his" and "him." The reason for this

[16] While this practice does not meet the ideal of contemporary reflexive anthropology (e.g., Fabian [1983]2002), I take it as a necessary evil.

choice is that all PKW project founders and the majority of the participants in these groups are men. While this state of affairs is problematic (as many of these men testify), it is not unique to PKW. A persistent gender imbalance can also be observed in the high-tech industry and in similar volunteer settings such as that of the American group Code for Boston. Some academic authors have rhetorically responded to this exigence by using only feminine pronouns. I refrain from replicating this practice not only because of its empirical inaccuracy but also because it has very little impact on the problematic reality it seeks to remedy. The discussion and analyses that follow thus use the masculine form as a generic placeholder for all participants.

Finally, the presentation of data is based on the following format:

Excerpt number	Excerpt catalogue details	
Line number	Name	Thematic unit of relevancy

Fig. 1. Transcription Conventions

The serial number of each data excerpt appears at the top left corner. Additional catalogue information is provided next to this number, e.g., Interview (1/10/2016). Each subsequent line of text is marked by a number (i.e., 1, 2 … n). Pseudonyms of interviewees or participants appear next to this number only if the data represents two or more speakers. The separation of data content to distinct lines is meant to reflect transitions between thematic units of analytical interest. If a line-by-line analysis is in order, then one could expect to see a micro-segmentation of

participants' actions and thoughts. In other cases where data is presented primarily for illustration purposes, thematic units are organized into larger chunks of text.

Other compositional choices are the usage of italics in defining theoretical concepts (e.g., *communication practice*); the usage of double quotation marks in citing native terms (e.g., "code donation"); the usage of single quotation marks in invoking idiomatic or metaphoric expressions (e.g., 'too many cooks ruin the soup'); and the usage of capital letters in creating/invoking acronyms (e.g., The Public Knowledge Workshop or PKW) and in denoting officially named locations and organizations (e.g., Tel Aviv, Code for Boston, Open Knesset).

Actual Fieldsites

In 2015, PKW would typically hold two weekly development meetings located within a high-tech tower rented by the Google corporation at the city of Tel Aviv (Figs. 2-4), and to a lesser extent at a high-tech "hub" in the National Library of Israel at the city of Jerusalem (Figs. 5-6). While both spaces were active every Monday for several hours (usually between 20:00 and 00:00), the meeting place in Tel Aviv functioned as the de facto headquarter of PKW.

Aside from the development meetings, PKW administrators used to organize occasional "hackathons." These larger events are locally defined as intense development meetings that encompass an entire weekend. PKW hackathons occur on an irregular basis and

are designed to generate enthusiasm around the production of civic websites while attracting new volunteers to the eKnights.[17]

Fig. 2. The Google Tower

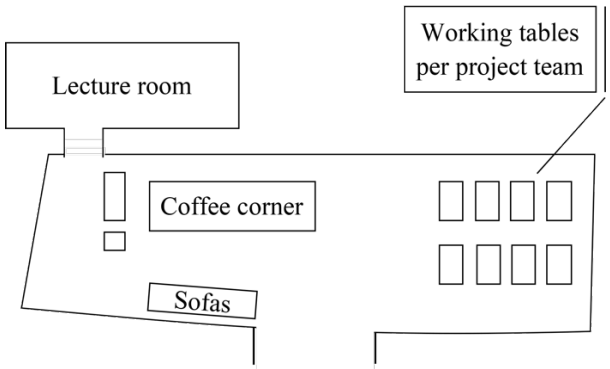


Fig. 3. The Meeting Place in Google Tower

Fig. 4. Participants in a Project Team at Their Table

[17] More generally, the term "hackathon" refers to a large communication situation in which amateur and professional technologists gather together to engage in activities of software and/or hardware development over a weekend. While the origin of this type of situation can be traced to the carnivalesque "hacker conferences" of the 1990s (Coleman 2013), today the notion of hackathon is used as a rather ambiguous term that covers several different kinds of occasions. Many hackathons are organized by large commercial firms that harness the creative energies of external developers for the experimentation and testing of new technological systems. Other hackathons are coordinated by groups of community organizers who may or may not be related to a specific social organization, an open source project, or a social or technological cause. In many cases, the metaphorical association of "hackathons" with "marathons" is used to emphasize the competitive nature of the situation. For a linguistic anthropological study of social interactions among hackathon participants see Jones et al. (2015). For a more comprehensive discussion of the political and economic vistas of open government data hackthons, see the excellent ethnographic work of media anthropologist Lilly Irani (2015).

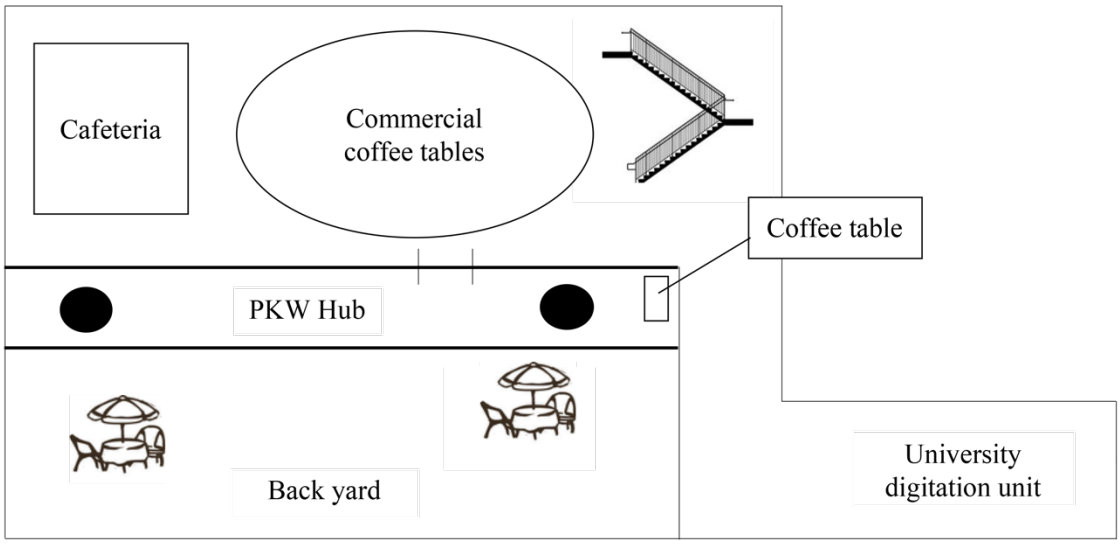


Fig. 5. PKW's Hub in Jerusalem

Fig. 6. Inside the Hub at the Time of Hackita02

Finally, one of PKW participants — a professional web developer and programming teacher — started in 2014 a personal initiative that he called "hackita," an amalgamation of the English word "hack" and the Hebrew word "kita" or class (as in classroom). Hackita is an eight-week programming course that focuses on the technological means by which PKW eKnights produce

their civic websites. To my knowledge at the time of this writing, hackita program ran only twice. The first round was in 2014, and the second in 2015 (at the time of my fieldwork). The second round of hackita took place in PKW hub at Jerusalem, and was whimsically named hackita02 after this city's phone prefix. The two participants[18] who ran the program performed a strict screening process with the aim of creating a group of high-quality students. The group, which eventually consisted of twenty-three people, met every Wednesday between 10:00 and 18:00.

Data Collection

I first approached the Voluntary Association of The Public Knowledge Workshop in 2015 through online communication with the group's "community coordinator." Following this person's request, I published a short preamble in PKW online forum to inform participants of my intent to conduct research and ask for their consent to be observed and documented. This preamble was ignored by everyone except the community coordinator who interpreted the group's silence as an uninterested acceptance. I was thus given informal permission to attend and study the development meetings.

By serendipitous coincidence, the program of hackita02 had just begun when I arrived at Israel in November 2015. Gaining entry into this second fieldsite was relatively harder. I was required to write a formal application letter for this purpose

[18] The person who ran the hackita program recruited one of PKW eKnight project founders to assist him in this voluntary work.

and to pass an informal telephone interview with the person who created and managed the program. Given that the fulfillment of these requirements took weeks, I could not observe the first two meetings where hackita students were divided into small project teams that would give them social frames of reference for the remainder of the program.

After this initial delay, I had full ethnographic access to the development meetings and to hackita02 meetings in Jerusalem. However, I was not able to learn a great deal by merely observing these situations. Generally, I saw people working silently at their computers. There are no 'do not disturb' signs in these arenas of software production, but one easily perceives the social imperative in this context, through body language and the absence of conversation, that those at work should not be disturbed. One of the more visually apparent non-verbal announcements of this imperative is the participants' tendency to wear headphones while executing individual programming tasks, which encourages visitors who cannot program or immediately integrate into the groups in some other way, to leave.

In total, I attended development meetings in Tel Aviv ten times for three to four hours per visit, and the hackita02 meetings in Jerusalem six times for six to eight hours each. The embarrassment of pretending to take fieldnotes at these semi-industrial locations paid off in two ways. First, it allowed me to merge organically into the background similarly to other objects present in these situations. Secondly, it provided me

with opportunities to participate in informal activities such as a small open government data conference, a variety of public lectures, and other meetings between PKW eKnight founders and hackita02 students, all of which improved my sense of how the teams work.

My naturalization into the social setting of the Jerusalem hub and the generosity of hackita02 initiator who genuinely attempted to accommodate my needs, allowed me to use audio recording devices on five different occasions. Two of these occasions were meetings between hackita's students and the PKW project founders mentioned above. The first meeting was a forty-minute talk that a leading participant in one eKnight gave to hackita02 students within the context of a classroom lecture. The second meeting was a one-hour panel conversation between two eKnight founders, three other interested parties who ran experimental projects in PKW, the group's community coordinator, hackita's program initiator, and all of hackita02 students.[19]

The open government data conference I observed in Tel Aviv was also audio recorded. Given that this event was open to the public, I did not seek permission to record, but did set up my recording device in a location visible to all participants (i.e., on top of my field notes book that rested on my knees). This three-hour event consisted of short talks by PKW CEO, one member

[19] The other three events I recorded were formal programming lectures by hackita's initiator. These events have small relevance to the present study and are therefore excluded from the primary corpus.

of the group's Board of Directors, and two regular team members in one of the group's eKnights. Beyond these opportunities for audio recording, my naturalization into PKW volunteer setting provided me with crucial access to individual participants with whom I could conduct more in-depth interviews. The results of these latter efforts are reported below.

In-depth Interviews

It is a commonplace assumption among ethnographers and cultural discourse analysts that people are quite willing to talk about themselves in social situations when given the opportunity, especially where others treat them as figures of authority whose opinions and actions are of utmost importance. Unfortunately, this assumption did not hold true in PKW development meetings whose participants operated under the condition of time scarcity. While no one said so explicitly, the message communicated to me was that my expectation that volunteers will sit and talk with me about their projects at a time when they could actually develop these initiatives was unreasonable. As my research progressed, it became clear that this indirect message involved certain normative assumptions about proper conduct in the group's arenas of software production.

Under these unusual restrictions, I managed to talk with ten individuals, eight of whom were active or veteran participants in one of the eKnights. The other two were members of PKW administrative body. Before providing more details about

these individuals, it will be useful to specify the organizational categories to which they belong.

Categories of Participation

The Voluntary Association of PKW consists of two distinct parts: an administrative body and a community of eKnights. The historical process that led to this duality is elaborated in chapter 4. For now, it is important to understand that PKW administrative body functions primarily as a service provider of the different project teams and therefore has very small managerial power over these other groups. By Israeli law, any registered Voluntary Association must institute a number of entities that include a General Meeting, a Board of Directors, and an Audit Committee. In addition to these institutional entities, PKW hires a Chief Executive Officer (an official requirement) who performs most of the necessary administrative work, and one community coordinator whose job is to assist the CEO and support the different project teams.

In the meeting space at Tel Aviv, each project team would have a "regular table" (see Fig. 4 above).[20] As independent units of sociality, these teams organize around the developers who stated the projects. For the most part, these project founders have working relations with at least one highly committed

[20] The notion of "regular table" in PKW volunteer setting is the product of a widespread convention rather than an official rule. If the regular table of a given team is found to be occupied, the participants will move to another table (although such a scenario is relatively rare).

volunteer who can be considered as a co-developer in Raymond's ([1999]2001) terms. A second and wider circle of participation consists of slightly less committed and productive volunteers who regularly attend the development meetings or otherwise participate in the projects online. The number of these individuals varies across project teams but is relatively small (one to five persons). Taken together, the project leader, co-developers, and regular members of each such group, comprise what I refer to as a "core team."

At the time of my fieldwork, I could clearly see six such core teams. Five teams regularly attended the development meetings, and one team worked primarily online. The largest and most celebrated teams were two groups whose members instituted PKW as a Voluntary Association in 2011. The first eKnight called Open Knesset develops a civic website that tracks and analyzes the bills and votes of the Israeli parliament. The second eKnight called Open Budget (and later Budget Key) monitors the parliamentary processes by which the Israeli state budget is distributed. The eKnight whose members work online is the Open Urban Building Scheme that makes public and private building plans accessible to internet users. The three other eKnights are AnyWay (tracks and analyzes patterns of car accidents), The State Square (tracks activities of Israeli parliament members on Facebook), and Open Train (calculates patterns of delay in Israeli train transportation). Altogether, these six core teams function as a source of attraction for a much wider circle of

occasional volunteer participants who are known among the more committed team members as "code donors."[21] Some of these donors temporarily attend the development meetings, while others offer their contributions online.

In addition to the core teams and the occasional code donors who support them, there is another circle of participation that consists of "free floaters" (my term). While most of PKW participants are dedicated to a particular eKnight (regardless of their degrees of commitment), free floaters lack such an affiliation. I do not know how many such individuals were active at the time of my fieldwork, but I believe that the number was relatively small. Wearing the "colors" of a specific "team," as one free floater phrased it, is the norm in PKW volunteer setting.[22]

[21] The terms "code donor" and "code donation" are used in relation to anyone who contributes to the development of the group's eKnights. At the same time, core team members are also associated with other terms such as "project leaders" and "volunteer(s) in the project." While the term "code donor" is in general use, these other terms are more common in the meta-pragmatic discourses by which core team members speak about their organizational identities. In the interest of analytical clarity, I shall therefore use the term "code donors" in reference to occasional participants who are not recognized as core team members.

[22] Note that lexical items such as "color" and "team" are the subject matter of CuDA as they demonstrate the reliance of participants on local means of expression in signaling who they are and how they are related to each other. Within the framework of the present study, the terms "code donation," "to do/make code" and "doocracy" are examined as local manifestations of PKW participants' economic rationality, and thus as constitutive elements of the group's technosocial system and system of governance.

List of Interviewees

The seven developers who agreed to in-depth interviews were one regular team member, three project founders, one student of hackita02, and two free floaters. The details of these interviews are summarized below.

Participant role	Project affiliation	Duration
Regular team member	Open Train	0:56:00
Project leader	Open Urban Building Scheme	1:06:00
Project leader	The State Square	1:43:00
Project leader	Open Knesset	1:15:00
Hackita02 student	N/A	1:05:00
Free Floater	N/A	1:17:00
Free Floater	N/A	0:30:00
SUM		~8 hours

Table. 1. In-depth Interviews with PKW Volunteers[23]

Given that the Israeli open government data scene is relatively small, curious readers will be able to guess the identities of the interviewees listed above. The following elaboration is therefore designed to respect the privacy of the participants at issue. Its only purpose is to assure the reader that the analyses developed in the data-based chapters represent diverse perspectives of group members whom I came to know in the field to some degree of acquaintanceship. The reader does not need to recall any of the characters at issue as they do not play any role in organizing the text. Moreover, the reader should keep in mind that this list of interviewees is essentially partial as the analyses cite the voices of other participants whom I

[23] All of these interviews were logged and transcribed in full.

recorded and/or interviewed in the field. While maintaining the consistency of participant roles throughout, the discussion intentionally mixes pseudonyms and gender identities. While this procedure is imperfect on both methodological and ethical levels, it is the best I could think of.

With this ethical consideration in mind, let us note that the first interviewee was a young woman who participated in an eKnight for more than a year and was also a former student of one of my teachers at the University of Haifa. This social affiliation allowed me to contact her prior to my arrival at Israel and ask for her assistance in my preparations for the study. The interview took place at the Google Tower in Tel Aviv just before my initial entry into the field. This person also shared with me an unpublished B.A. thesis she had written about her participation in PKW.

The second interviewee was the project founder of The Open Urban Building Scheme. This person ran a profitable high-tech company and was a well-known and respected figure in PKW volunteer setting. Much of my basic knowledge about the group's organization of production comes from this conversation.

The third interviewee was the founder of The State Square eKnight. This person was a student in the first round of hackita program where he conceived the idea of tracking the activities of Israeli parliament members on Facebook, and acquired the basic technological skills of a web developer. After completing the eight weeks training course of hackita, his initiative was

accepted into PKW community of practice. Having this conversation after gaining basic knowledge about the group's organization of production component of technosocial system from the previous interviewee allowed me to pose more specific questions about that organization of production's component of system of governance.

The fourth interviewee was the founder of a personal project that ultimately evolved into Open Knesset, PKW historical precursor. Naturally, this interview would be invaluable to the present study but since this person officially announced his withdrawal from Open Knesset by the end of 2014, he was not present at the group's development meetings at the time of my fieldwork. Knowing that such an in-depth interview could help refine my analytical claims, I chose to save it for a later stage of fieldwork. By 2017, I was much better prepared to conduct this interview and was fortunate enough to gain this person's full cooperation. The interview was conducted remotely from the U.S. through the Skype telecommunication application.

The fifth interviewee functioned as the leader of the project team in which he participated as a hackita02 student. To the best of my knowledge, this person was the only hackita02 student who integrated into PKW volunteer setting. I was made aware that he adopted an abandoned project called Open Pension with the intention of recreating it from scratch, but it was not completed during the period of my fieldwork, and I have no further information regarding this enterprise.

The sixth interviewee was a graphic designer who contributed to most of the organization's projects in various capacities. The macro-social perspective of this individual, who freely moved and operated in PKW volunteer setting from its early beginning, was helpful for my understanding of the organizational challenges and group dynamics pertaining to this particular social scene. At the time of my fieldwork, this person also assisted hackita's organizers and was an active participant in the project that the fifth interviewee led.

The seventh interviewee was a former member of PKW Board of Directors and a programmer who had attempted to create several different projects in the group's volunteer setting. When I first entered the field, the community coordinator of PKW assigned me to that individual due to his shortage of workforce personnel. Given the marginality of this participant's initiative and its highly political nature, I maintained a distance from it. In return, this person gave me a relatively short and disinterested interview. Nevertheless, I found his perspective useful for the validation of claims made by other interviewees.

In addition to these seven interviews that focused on the perspectives of participants in PKW eKnights, I also interviewed three past and present members of the group's administrative body. The technical details of these additional interviews are specified below.

Participant role	Duration
Ex community coordinator	01:25:00

Ex community coordinator	00:53:00
Chief Executive Officer	00:55:00
SUM	~3.5 hours

Table. 2. Interviews with PKW Staff[24]

The eighth interviewee served as the first community coordinator for PKW and had a job in the British open government data organization Public Knowledge at the time of my fieldwork. Due to this person's activist background and participation in PKW during its foundational years, s/he was particularly helpful in explaining the group's internal politics and its civic ideology of the common good.

The ninth interviewee, an anthropologist turned information analyst and web developer, was a student in hackita02 and a regular member of Open Knesset at the time of my fieldwork. This highly articulate individual provided me with a comprehensive cultural interpretation of PKW volunteer setting. While taking his perspective with due skepticism, I found many of his ideas and suggestions helpful. For example, Steven Weber's book The Success of Open Source (2004) that served as a major source of inspiration for this study was suggested to me by this person.

The tenth and last interviewee served as PKW CEO until very recently. Of particular significance were her insights about the nature of the relationship between the group's eKnight founders and official administration.

[24] All of these interviews were logged and transcribed in full.

Interview Data Collection

On a methodological level, I approached the interviews listed above as open-ended conversations whose range of topics could be narrowed down as the study progressed. I started each interview with a request for a biographical story that would begin wherever the interviewee deemed appropriate and would then conclude at the chronological moment of his or her integration into PKW. After the completion of this story, I would pose general questions about the group's mission statement, organizational features, and communication practices.[25] This procedure allowed me to eliminate topics less germane to the study and sharpen my focus in subsequent interviews on more relevant matters. I repeated this process throughout the study so that by the end of the last interview with Open Knesset's founder I felt comfortable that all necessary data had been collected to complete the analysis presented in this iteration of the study.

Virtual Sites

In addition to the actual settings of PKW development meetings, hackathons and hackita program, the group maintains an official website in which some participants will occasionally publish brief essays; an online forum that provides an historical documentation of the group's public conversations and debates; and a wiki website that provides information about the different

[25] Questions about the communication practices of project team members were guided by Hymes's (1972a) SPEAKING model.

eKnights and other related technical and social topics. While these resources were designed for general use, each project team employs specific media for communication to coordinate its activities. These media typically included email lists, smartphone applications such as WhatsApp, and team collaboration software such as Slack and Trello, among others; however, these carry a marginal role in comparison to the GitHub platform.

A more comprehensive examination of GitHub is presented in chapter 6, but for the moment, the key detail of this online platform is that it functions as a software production arena that provides programmers with a space called "repository" where eKnight source code can be maintained along with a variety of tools and online services for technical and social collaboration.

It is crucial to understand that GitHub's existence represented a primary condition of possibility for PKW technosocial system of product development. Each of the group's eKnights had a source code repository on this platform, and this allowed core team members and occasional code donors to participate in the production process from any geographic location where they had an Internet access, which in turn meant that participants did not need to attend the development meetings or even meet each other in person in order to collaborate.[26] Conversely, the participants who did attend the meetings also

[26] For example, a central contributor to The State Square project who lived in the U.S. worked remotely with his team members for more than a year before meeting them in person.

needed to connect to their online repositories on GitHub in order to collaborate. In fact, I found that interacting with a project team through GitHub was not only a possibility but also a preferred mode of engagement. At the early stages of my fieldwork, GitHub struck me as a counterintuitive communication platform for volunteers to meet each other and establish working relationships. But, over time, I began to recognize its sensibility through a noticeable contrast between the ways team members received visitors who had already contribute, and those who had not. It was this trend that shed initial light on the economic rationale that lay behind virtually all participant behavior in this volunteer setting.

Taken as a contemporary fieldsite for virtual anthropology (Boellstorff 2008), a project repository on GitHub functions not only as a place where participants can interact and work together, but also as a usable historical archive of such joint labor. Any ethnographer who seeks to study such projects thus has full access to every line of code that each participant has ever uploaded to the system, as well as to the online discussions that revolved around the acceptance of any such code contribution. While much of this history is technical, a person unfamiliar with computer code can still discern the social character of the working relations that he or she observes. And while the technical discussions on GitHub provide only a glimpse into the more complex natural histories of the different project teams,

this glimpse carries significant value to ethnographers studying these types of projects and communities of practice.

Secondary Data

In addition to the production activities by which PKW project team members run their eKnights on GitHub and the meta-pragmatic discourses through which some of them talk about such activities, this dissertation's corpus includes four secondary datasets. The first dataset consists of video recorded public talks and interviews by PKW participants. These materials can be found in the group's websites or in social media networks such as YouTube and Facebook. I collected and documented sixteen such video materials of which two were particularly relevant to the present study. These materials are public talks by one of PKW founders, a web developer with a considerable reputation in the Israeli high-tech and open source software scenes.[27]

The second dataset consists of twenty-three excerpts extracted from PKW online forum. I used these excerpts for purposes of triangulation; these are not cited in the analysis itself. The third dataset serves primarily as an analytical aid, consisting of two separate interviews with personal associates, each of whom carry significant expertise in the high-tech industry. The first associate is a software engineer who works in a successful high-tech company in Israel, and the second is an

[27] These two talks were logged and transcribed in full.

Israeli academic researcher who holds a PhD in computer science. These two interviewees helped me to understand the computer means of production by which PKW participants develop their civic websites, as well as a variety of other related technical, theoretical and social issues. Much of my description of the eKnights' technosocial system in Chapter 4 was examined and approved by these two experts.

The fourth dataset is based on a variety of published texts, mostly books, about Open Source Software (OSS) and other related technological issues. Most of these materials are mentioned in PKW wiki website and were explicitly suggested to me by several different participants and interviewees. These include the science fiction book The Cyberiad (1965) from which the term eKnight is derived; Eric Raymond's influential book The Cathedral and the Bazaar ([1999]2001) that documents the OSS culture of the mid-1990s from the perspective of a native developer; and Journalist Steven Levy's book Hackers: Heroes of the Computer Revolution (1984) that chronicles the early emergence of that culture around MIT in the 1960's; to name but a few. A review of these materials provided some degree of contextual understanding that appears in this dissertation, but not specifically enough to directly cite or reference the original texts.

Comparative Data

The corpus that incorporates the materials reviewed above is complemented by a comparative corpus that includes three

datasets. These datasets derive from my earlier fieldwork among members of the American civic group Code for Boston (CFB) in the city of Cambridge MA, and were preliminarily analyzed for the purpose of a comparative analysis with PKW development meetings.

CFB is a local branch or "brigade" of the larger organization Code for America that operates on a national level, and it has significant parallels with PKW. CFB participants associate themselves with specific project teams, rely on the GitHub platform for the development of their software products, meet on a weekly basis in a social gathering that they call "hack night," and engage in yearly hackathon events. And yet, there is a significant difference between the two groups: Whereas CFB resembles a loosely organized social club, PKW development meetings take the form of a semi-industrial site of software production. Analytically, this difference allowed me to determine the boundaries of PKW eKnights' organization of production and the degree to which this form of social organization relies upon culturally specific assumptions and normative arrangements.

The first dataset I used for this comparative purpose consists of in-depth interviews with CFB "brigade captain," his wife who served in an administrative role, the leader of the two most successful projects initiated in that social setting (cited in excerpt 1), a representative of Cambridge city hall who participated in one of the group's hackathons, and the official

Brigade Program Manager for Code for America.[28] The second dataset is based on ethnographic fieldnotes taken in CFB hack nights, and on video footages taken in one of the group's yearly hackathons.[29] The third and last dataset is based on materials collected from the GitHub repositories of the two aforementioned projects.[30] More specific details on this comparative data are provided in chapter 6 where it is used.

Concluding Remarks

This review of the study's corpus and the various ethnographic tools by which the data were generated is meant to provide a partial answer to the epistemological questions of how I came to know what I believe myself to know, and, hence, to what extent the reader should trust this knowledge.

---

[28] The average duration of these interviews is fifty minutes. While all the interviews were logged, I transcribed only relevant excerpts for the comparative analysis presented in chapter 6.

[29] All the hack-nights were documented based on handwritten field-notes. The hackathon at issue is the 2015 National Day of Civic Hacking (NDCH). Within the framework of this event, I videotaped a particular project team that developed a website called Need Now whose purpose is to provide homeless people with information about relevant city services and shelters. The project was proposed and initiated by the representative of Cambridge city hall I interviewed. This form of initiation derived from the definition of the 2015 NDCH as an effort to support other government agencies and social organizations. All in all, I recorded more than eight hours of people sitting and working in front of their computer screens while engaging in occasional conversations. Most of these recordings were logged but not transcribed.

[30] These projects are the Need Now initiative I observed in the 2015 NDCH, and MBTA Ninja that employs a crowdsourcing technology to provide users of the Boston subway with means to report on real-time delays in local train transportation.

This is not the place for a comprehensive discussion of the phenomenological and hermeneutical foundations of ethnography and how they relate to the quantitative and experimental models used by microeconomists. Suffice is to say that I made significant compromises in terms of the claims I wanted to make relative to the partiality of the data I had, and that these compromises reflect an ongoing and systemic consideration with regard to the claims I do make in the data-based chapters.

More generally, I find myself very much in agreement with American sociologist Robert Zussman (1992:231-232) who proposed ethnography as one, and perhaps not the best, approach to data collection in the social sciences. While not necessarily the best approach, ethnography is the preferred methodology of Cultural Discourse Analysis (CuDA) — a research framework that I was trained to use and to whose usage I am committed. My application of ethnographic tools such as non-participant observation and naturalistic audio/video recording, in this study, is therefore not an attempt to challenge the traditional methodologies of microeconomics. Rather, the application of these tools is meant to complement those other approaches in new and creative ways. This intention to expand methodological and analytical horizons is further elaborated in the following chapter.

CHAPTER 3

METHOD

Introduction

The overall aim of this dissertation is to show how the microeconomic conception of human agency as the capacity of all people for rational action and thought may help cultural discourse analysts understand the underlying conditions of emergence for at least some communication systems that may become the subjects of their scholarly inquiries.

The theoretical rationale by which the study attempts to accomplish this aim can be stated as the following proposition. If the particular communication system constitutive of PKW organization of production manifests the human nature principle of utility maximization, then any local group of programmers who seek to organize for production purposes under similar conditions of time scarcity and lack of command ability will necessarily arrive at some version of that same system.

To support this proposition, the analysis uses a two-pronged strategy. On the one hand, it employs the conceptual framework of Cultural Discourse Analysis (CuDA) as means to describe and interpret the particular communication system constitutive of PKW organization of production. On the other hand, it borrows ideas and concepts from the field of

microeconomics to infer efficiency optimization as the raison d'être behind this communication system.[31]

Analytical chapters 4 to 8 conduct these descriptive-interpretive and explanatory moves vis-a-vis concrete obstacles or 'problems' that any group of programmers will face if they are to work together over time, across geographic distance, and without remuneration. The first problem concerns the possibility of corporate members to produce complex software under the condition of lack of command ability. The second problem concerns the economic risk of corporate members who need to invest some of their scarce leisure time in recruiting volunteers who may abandon their tasks and exit the projects whenever they see fit. The third problem concerns the general tension between autocracy and democracy in voluntary corporations. The imperative of the analysis is to show that the communication practices through which participants in PKW eKnights have confronted these three problems are methods that any rational actor could have arrived at in both theory and practice.

Given that these analytical pairs of a *problem* and its *method of solution* can be best formulated in terms of CuDA, the

[31] Using CuDA in the first move not only reflects my intention to contribute to its explanatory framework, but also my belief in its analytical superiority. Unlike many other cultural-interpretive approaches in the social sciences whose practitioners focus their analyses on the abstracted normative and symbolic orders of presupposed societies, CuDA is designed especially for the in-situ analysis of the pragmatic and meta-pragmatic practices by which group members materialize, represent and converse about such orders of signification.

following discussion starts with a brief consideration of this conceptual framework. The discussion then formulates the relationship among economic rationality, communication practice and social organization that the analysis seeks to defend. Finally, the discussion explains the procedure by which the analysis achieves its empirical and theoretical ends.

Conceptual Framework

On the foundational level of social inquiry that Carbaugh and Hastings (1992) call "basic orientation," this dissertation starts with the proposition that if communication practices constitute a part of all human social life (Carbaugh 1995), then such practices must also constitute parts of the technosocial system and system of governance by which participants in PKW eKnights create their organization of production.

Within CuDA, the term *communication practice* refers to a descriptive category that incorporates three social units. The first unit, *communication act*, provides ethnographers with a means to describe how people perform individual actions within a given social situation (Carbaugh 1989:98), and the variable ways in which all others present interpret that performance (Carbaugh 2007b:2-4). The second unit of a *communication event* provides CuDA scholars with means to describe culturally named or recognized act sequences that require two or more individuals for their performance (Carbaugh 1989:99), and involve culturally bounded aspects of social life which have a beginning and ending

(Carbaugh 2007b:2-4). Finally, the unit of *communication style* refers to the systemic ways by which group members organize and select between alternative acts and events (Carbaugh 1989:100). When people make stylistic choices, they show awareness that there is more than one way to engage with an activity. This kind of awareness can grow out of intercultural contacts or other encounters with difference (Katriel 1986).

Within CuDA, communication acts, events and styles are conventionally defined as referents of symbols and symbolic terms for communication practices. For example, some participants in PKW eKnights refer to their primary event of software production by the symbolic term "terumat cod" (i.e., code donation), and to the communication style that defines their development meetings by the term "la'asot cod" (i.e., to do/make code). Following Carbaugh (1995), I propose to conceptualize each such Symbol For Communication Practice (SFCP) in three interrelated ways; first, as a *denotative symbol* that forms a part of an *activity system*; second, as a *connotative symbol* that forms a part of a *value system*; and third, as a *regimented symbol* that forms a part of a *rule system* (Fig. 7).

Taken together, the activity, value and rule systems that converge upon a specific cluster of SFCPs can be defined as a *cultural discourse* in Carbaugh's (2007a) original sense. The conceptual entity of cultural discourse is the object of any CuDA research that seeks to examine the role of human communication in processes of sociocultural formation.

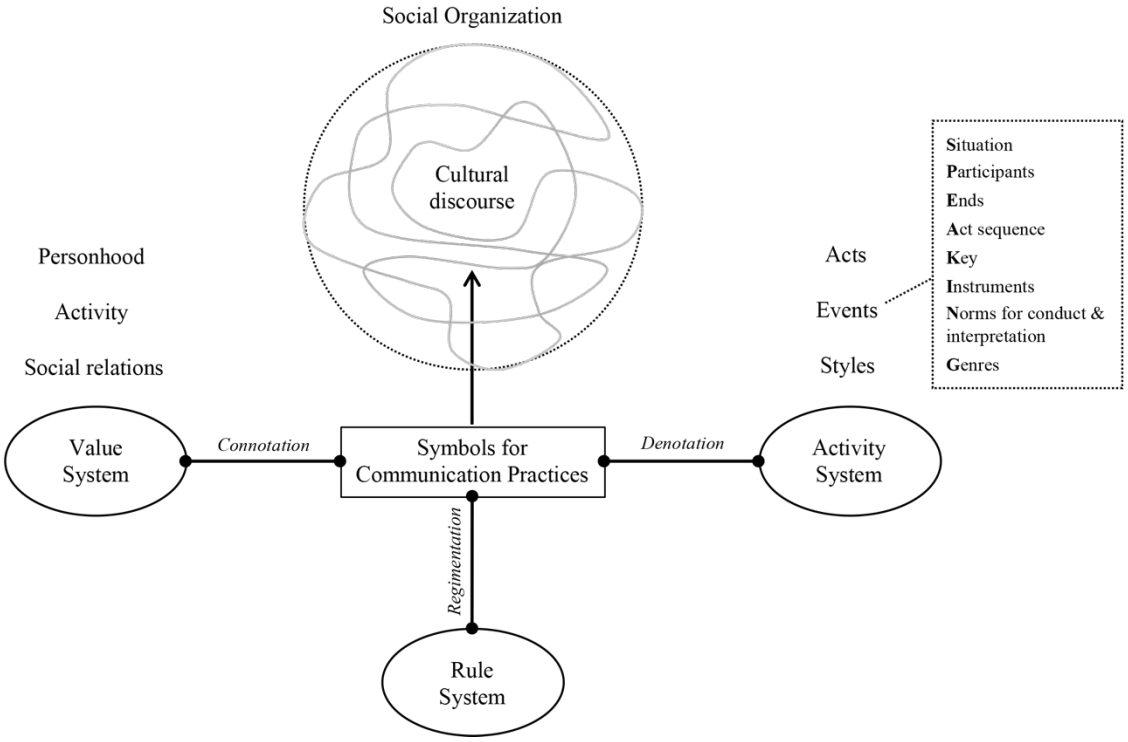


Fig. 7. The Anatomy of Cultural Discourse

In their capacity as elements of an activity system, SFCPs denote the communication acts, events and styles by which individuals constitute social forms of organization. Among these practices, the preferred unit of analysis is that of communication event due to the important role it plays in the consolidation of core cultural assumptions and beliefs. The analysis of any communication event is in effect an examination of the contextual constraints that give it shape and credence (Becker 1995:419-424). Hymes (1972a:58-65) famously conceptualized these constraints under the SPEAKING acronym, which refers, respectively, to contextual dimensions of the social Situation, the Participants and their Ends, the Act sequence that structures the event and the event's social Keying,

the communication media or Instruments that participants use and the Norms of conduct and interpretation they follow, and, finally, the Genres to which parts of the event's act sequence may classify as members.

In their capacity as elements of a value system, SFCPs orient the acts, events and styles they denote toward shared values and beliefs. When group members use such SFCPs, they not only legitimize the actual practices in which they are engaged; they also create transient loci for reflexive talk about these practices and their value-laden meanings. Within CuDA, the connotative meanings of SFCPs are conventionally classified into three *discursive hubs*, namely personhood (i.e., being), activity (i.e., acting) and social relations (i.e., relating).[32] When group members explicitly activate one of these discursive hubs, they may convey implicit meanings that can be said to *radiate* from that specific hub as well as from the other two hubs (Carbaugh 2007a; Carbaugh and Cerulli 2013).[33]

Finally, in their capacity as elements of a rule system, SFCPs are normative templates or *protocols* for the correct performance and interpretation of social action. To the extent

[32] More recently, Carbaugh (2007a) added to this list the hubs of *affect* and *dwelling* that have only small relevancy to PKW discourse of work.

[33] To deal with this complexity, Carbaugh and Cerulli (2013) proposed a two-step procedure for the interpretation of SFCPs. In the first step, the analyst draws on the actual terms and phrases of participants to formulate lists of *cultural propositions* proper to each discursive hub. In the second step, the analyst infers the *cultural premises* that permeate these propositions and of which group members usually have little to no awareness.

that such protocols guide the conduct of communication events, they are entirely congruent with Hymes's (1972a) constraint of Norms. However, the normative protocols of a group's discursive rule system have a broader descriptive scope as they apply to all three social units as well as to rules pertaining to specific situations and gatherings (Carbaugh 1990). Insofar as such normative protocols are known, shared and specifiable by all group members, these individuals will not only employ the protocols in specific situations and events, but will also provide reflexive commentaries about them in the form of explanations, justifications and reflections (Carbaugh 1995). Such meta-communicative commentaries usually revolve around common knowledge about "who has the right to do what, and with which, and to whom, when, where, and how" (Schneider 1976:199).

Working Assumptions

The present study applies the CuDA framework to the case of PKW on two basic assumptions. The first assumption states that there must be a local discourse of work by which participants in the eKnights constitute parts of the technosocial system and system of governance formative of their organization of production. Otherwise, these individuals would have not been able to have a common sense of "who they are, how they are related to each other, how they feel, what they are doing, and how they are situated in the nature of things" (Carbaugh 2007a:168). While the entity of cultural discourse is constitutive of social

organization only in part, no social organization could exist or function without it (Carbaugh 1995).

The second, derivative, assumption states that insofar as PKW participants use a local discourse of work, this discourse must consist of one or more SFCPs that operate simultaneously as (i) denotative symbols in an activity system whose referents are communication acts, events and styles of software production; (ii) connotative symbols in a value system whose semantic fields involve moral meanings about personhood, activity and social relations; and (iii) regimented symbols in a rule system whose purpose is to orient the production practices that the activity system denotes in accordance with the moral meanings that the value system connotes (Carbaugh 1995:285). Hence the theoretical connection among the three systems that together constitute the discourse of work that forms a part of PKW organization of production's technosocial system and system of governance.

Procedure of Knowledge

This study's contribution to CuDA lies in its attempt to show that human nature factors such as the capacity of all people for rational action/choice may account for the emergence of cultural discourses and may, therefore, be used as means to develop an explanatory mode of inquiry within that research program, a mode of inquiry that puts the active agency of individual actors front and center. The intention here is not to undermine the significance of communication to social theory. On

the contrary, this dissertation confirms that the theoretical significance of cultural discourses lies precisely in their prowess of mediation. While the entity of cultural discourse may not cause the social organization of which it is a constituent, it is a necessary condition for the operation of any such first cause. Cultural discourses thus have a crucial role to play in the development of any social theory.

This study presents its finding in the form of a thesis to be supported rather than in the form of a hypothesis to be tested due to the inductive process that led to its formulation. The first iteration of the draft in hand aimed to describe the cultural discourse of work by which participants in PKW eKnights constitute their organization of production. Only after the completion of that initial draft, it became clear that PKW discourse of work articulates and actualizes the principle of utility maximization in local terms, and that the organization of production described here is the most efficient one that any group of rational actors could have arrived at under the limiting conditions of time scarcity and lack of command ability.

These results led me to the following conclusions that together form the central finding of this dissertation: (i) the rational actions or choices of individuals who move and operate under an existential condition of scarcity may explain why a given cultural discourse came to be the way it is; and (ii) that, therefore, the emergence of cultural discourses can be partly explained by constant human nature factors and principles. The

loci of such principles of individual action and motivation can be considered as the *causal impetus* which is presumed for a given cultural discourse to be what it is.[34] Hopefully, this conceptualization would allow CuDA scholars to infer the logic of the specific discourses of work they describe and analyze if and when such an explanatory move is in order.

Operative Framework

To construct the finding[35] presented above, the study formulates a simple causal relationship among an explanatory variable, a response variable and a mediating variable (Fig. 8). The response variable, organization of production, is defined as an instance of social organization that consists of a technosocial system and a system of governance. The explanatory variable, economic rationality, operationalizes the utilitarian supposition that voluntary corporate members must have a common interest to maximize their individual utilities through the creation of the most efficient organization of production that can be created under constant conditions of time scarcity and lack of command ability. The general principle of utility maximization is thus defined in terms of *efficiency optimization*.

[34] I thank Donal Carbaugh for this conceptualization.

[35] The remainder of this discussion substitutes the notion of argument with the notion of finding so as to emphasize the empirical veracity of its analytical claims.

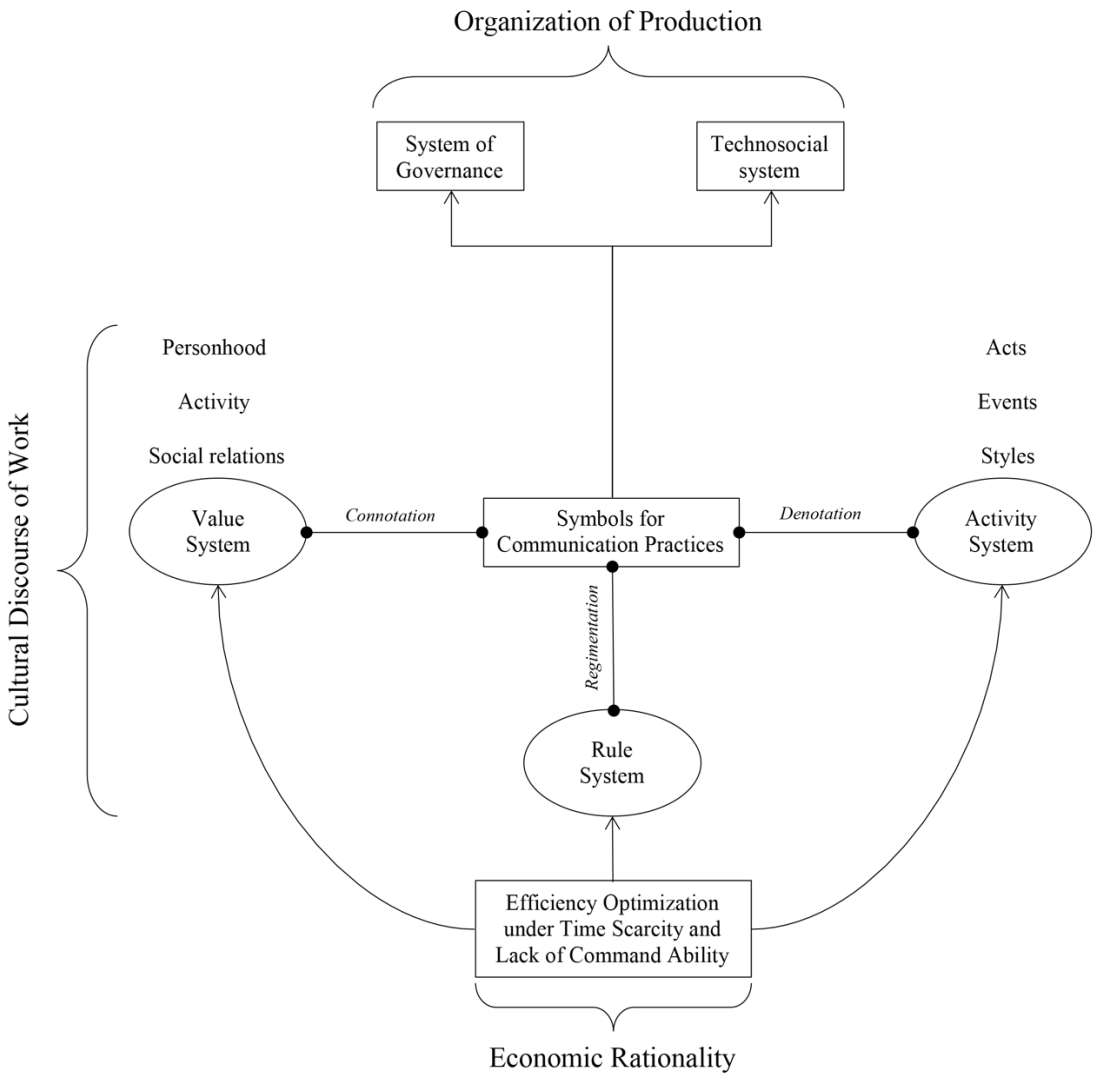


Fig. 8. The Thesis of Economic Communication

The mediating variable, cultural discourse, is the local communication system in and through which the causal relationship between economic rationality and social organization is articulated and actualized. That is, the conceptual entity of cultural discourse reflects the immanent social reality in which rational action/choice is getting accomplished.

Style of Argumentation

This dissertation is written only in a 'top-down' fashion. However, it is crucial to keep in mind that the finding reported here derives entirely from my prior and ongoing ethnographic inquiry. While the actual chronology of the research is rather complex, suffice is to say that my arrival at this finding is the product of five years of tedious work which involved mutually constitutive procedures of data analysis and theoretical inference. This delay derives, in part, from the fact that I was required to work against my own prior assumptions and intuitions. That is, my ongoing empirical examination was in effect an attempt to question the veracity of a homo oeconomicus. In the end, my commitment to the truth of the matter required me to show the exact opposite of what I believed myself to be able to show at the beginning of the research.

Chapter Outline

The analyses developed in the following, data-based, chapters are designed to support the theoretical model presented in Fig. 8 by focusing on the practical problems it appears to solve within PKW volunteer setting.

Chapter 4 opens the discussion with the two interrelated questions of technosocial coordination and scaling that ask, respectively: (i) How is it that volunteering software developers who choose to work together at the expense of their scarce leisure time, and thus under the condition of lack of command

ability, manage to coordinate their tasks over time and merge their individual contributions into functioning pieces of software? (ii) To the extent that these producers do manage to work together in collaboration, how do they also manage to maintain standards of software development and quality assurance in the desired context of growing numbers of participants?

The analytical sections of chapter 4 address these questions with an examination of the political institution known as "benevolent dictatorship" by which members of longstanding Open Source Software (OSS) projects such as the Python project overcame similar problems of coordination and scaling. The central claim of chapter 4 is that OSS voluntary corporations rely on the benevolent dictator institution as a condition for their possibility. To validate this claim and to provide necessary historical background of the group, chapter 5 explains how that institution was adapted by PKW founders — themselves participants in the original Python project. Chapter 6 then identifies the communication event of "terumat cod" (code donation) as an essential constituent of benevolent dictatorship, and shows how the contextual constraints of Situation, Participants, Instruments, and Act sequence in Hymes's (1972a) SPEAKING acronym shape this event in accordance with the economic principle of efficiency optimization.[36] The overall claim of

[36] This analysis does not address the contextual constraint of Norms in Hymes's framework due to its dependence on the orienting values discussed in the chapter that follows. Given that the norms at issue are part of the group's rule system, they are analyzed in chapter 8.

chapters 4 to 6 is that the economic rationality of PKW participants determines the creation of a technosocial system around the benevolent dictator institution through the mediation of the SFCP "terumat cod" in its capacity as an element in PKW discourse of work's activity system.

Chapter 7 leads the discussion to the essential dilemma of workforce recruitment among PKW benevolent dictators, i.e., how to reduce the risk of time expenditure in the activities required for the reception of new volunteers who may abandon their programming tasks at will. The analytical sections of chapter 7 work out this problem by showing how the economic interest of the group's benevolent dictators contributed to the sanctification of personhood traits such as assiduousness, proactivity and competence, and how participants in the different eKnights adhere to this valued model of personhood when they engage in software development activities. The overall finding of chapter 7 is that the economic rationality of PKW participants determines the creation of a valued model of personhood through the SFCP "la'asot cod" (to do/make code) in its capacity as an element in the group discourse of work's value system.

Finally, chapter 8 addresses the problem of participants in PKW project teams to make complicated decisions and resolve conflicts vis-a-vis the foundational tension between the regime-types of autocracy and democracy, and under the limiting conditions of time scarcity and lack of command ability. The analytical sections of this chapter show how the group's valued

model of personhood provides participants with means for the creation of a meritocratic system of governance that they call "dookratia" (do-ocracy). The overall finding of chapter 8 is that the economic rationality of PKW benevolent dictators determines the creation of a "dookratic" system of governance through the SFCPs "la'asot cod" and "terumat cod" in their capacity as elements in PKW discourse of work's rule system. This finding brings the analysis into conclusion as it effectively shows how a full-fledged organization of production (as an instance of social organization) may arise entirely from the self-interests of individual rational actors maximizing utility via locally emergent communication practices.

The concluding chapter 9 discusses some of the implications of this finding to the areas of CuDA and microeconomics.

CHAPTER 4

INSTITUTING BENEVOLENCE

Introduction

This chapter analyses extant literature on Open Source Software (OSS) with the aim of providing a preliminary solution to the first of three problems that any group of status equal programmers with weak social ties will face if they organize for purposes of work and production in a public place during their leisure time (as OSS developers most often do).

The problem at issue pertains to the organization of production component that I have called technosocial system as it concerns the ability of participants in voluntary corporations to coordinate their programming tasks in the desired context of growing numbers of volunteers. The solution to this technosocial problem of coordination and scaling is presented here in terms of the institutional entity known among OSS developers by the terms "benevolent dictatorship" and/or "benevolent dictatorship for life" (Raymond [1999]2001).

The theorization of benevolent dictatorship from prior scholarly literature is essential to the understanding of PKW communication practices of "terumat cod" (code donation) and "la'asot cod" (to do/make cod) on which chapters 6, 7 and 8 are focused. The present chapter can thus be read as an attempt to establish the background assumptions necessary for the interpretation of PKW discourse of work and the organizational

form it helps constitute. To accomplish this aim, the following discussion first presents the technosocial problem of coordination and scaling as encountered by members of any OSS corporation. The discussion then provides a brief consideration of the method by which this problem is analyzed and solved in the remainder of the present chapter.

The Problem of Coordination and Scaling

Any voluntary corporation that exists in the modern realm of leisure is constrained by the two conditions that I have called time scarcity and lack of command ability. The nature of these conditions and the relationship between them can be further illuminated in relation to the following passages by cultural anthropologist Victor Turner (1982:36-37, italics in original):[37]

> The cultural reality of leisure is […] influenced by the domain of work from which it has been split by the wedge of industrial modernization. Work and leisure interact, each individual participates in both realms, and the modes of work organization affect the styles of leisure pursuits […] Leisure time [as opposed to the modern working time] is associated with two types of freedom, "freedom from" and "freedom to" […] (1) It represents *freedom from* a whole heap of institutional obligations prescribed by the basic forms of social, particularly technological and bureaucratic, organizations. (2) For each individual, it means *freedom from* the forced, chronologically regulated rhythms of factory and office and a chance to recuperate and enjoy natural, biological rhythms again. Leisure is also (1) *freedom to* enter, even to generate new symbolic worlds of entertainment, sports, games, diversions of all kinds. It is, furthermore, (2) *freedom to* transcend social structural limitations, freedom to play . . .

[37] In this analysis, Turner draws on Durkheim's sociological project and on the work of French historian Joffre Dumazedier (e.g., 1968). The analytical proof that members of ancient traditional tribes did not have the same sense of leisure time as we moderns do is provided in the latter's work.

> with ideas, with fantasies, with words […], with paint, and with social relationships.

From here we learn that the condition of time scarcity in PKW volunteer setting results from the cultural opposition between work and leisure at the foundation of modern industrial society. To volunteer in PKW eKnights is to give up other recreational activities that might serve as mental respite from the type of work that modern actors need to perform in their professional capacities day to day. This not only explains how software production time in PKW eKnights is scarce, but also why participants in these voluntary corporations are constrained by the condition of lack of command ability. PKW volunteers are free from authoritative and hierarchical structures, especially from those found in their industrial workplaces, precisely because their practices of software production take place at a short duration of autonomy where each of them can only make decisions freely and perform work of his choosing.

This is where we find the problem that faced participants of PKW, and that constitutes the empirical puzzle of this dissertation. Insofar as the programmers who volunteer in the group's eKnights are rational actors, they will be interested in the creation of an efficient organization of production that would allow them to maximize their individual utilities under the condition of time scarcity. However, common sense tells us (as it must have told them) that such efficiency cannot be easily achieved under the condition of lack of command ability.

The present chapter addresses this problem by focusing only on its technosocial elements. Theoretically, it proceeds from the assumption that any two or more individuals who seek to create a voluntary corporation in the modern realm of leisure will find themselves in a situation where none has legitimacy to enforce his or her will as a constraint upon the actions of others (hence the condition of lack of command ability). Any technosocial system of product development that emerges in the modern realm of leisure must therefore involve the freedom of choice of participants to contribute to their respective projects however they see fit, and the logic for seeing their contributions as the sole criterion for their own membership in the groups.

One can thus see how the general condition of lack of command ability may lead participants in voluntary corporations to create a libertarian contract with two reciprocal norms: a norm of voluntary participation and a norm of voluntary selection of tasks (cf., Weber 2004:62). Participants in a voluntary corporation's organization of production need to display mutual respect for each other's individual autonomy not so much because they take such autonomy as a culturally sanctioned "sacred object" (Philipsen 1987), but because they are equally aware that none of them can force the others to do what he or she wants, a condition of lack of command ability that results from the opposition between work and leisure and its implications for individual choice and autonomy in modern industrial society.

Guiding Question

The puzzle of PKW organization of production can thus be reduced to two interrelated technosocial problems. The first is a problem of coordination that concerns the possibility of participants in OSS corporations to divide the labor and merge their individual contributions into functioning pieces of software. The second is a problem of scaling that concerns the ability of voluntary corporate members to create and maintain standards of quality in the desired context of growing numbers of workers, and can be best conceived in terms of the idiomatic notion that 'too many cooks ruin the soup.'[38]

Taken together, these two problems reflect a very basic difficulty that any programmer would encounter if he were to start a voluntary corporation within the modern realm of leisure. The mere fact of PKW existence, then, implies the operation of some institutional entity capable of enforcing a degree of organizational order upon participants in the group's eKnights. Accordingly, the question that guides the investigation in this chapter becomes: what is the nature of that institutional entity which is capable of organizing processes of software production in the modern realm of leisure, and, therefore, under the condition of lack of command ability?

[38] In more formal terms, the problem here is one of negative marginal utility whereby each addition of workforce (above a certain limit) reduces the quality of the product to the point where it ceases to exist.

METHOD

Given what we know about the typical OSS project, which I consider here as a larger category of voluntary corporations to which PKW eKnights classify as members, there are two probable candidates for the institution in question, each of which highlights different facets of OSS. The first candidate is the open source licensing scheme, which, according to Weber (2004:84-86), provides an important resource for the creation of any OSS corporate structure. The second candidate is the Anglo-American institution of land tenure whose tacit operation in OSS production was first indicated by software developer Eric Raymond ([1999]2001). While both of these institutions have been discussed in the literature, the precise relationship between them and the relative significance of each in shaping the OSS organization of production have yet to be decided.

The requirement to address this issue at the outset comes from the theoretical and analytical need to identify the organizational essence of any OSS corporation and of PKW eKnights thereof. By *organizational essence*, I mean the institutional entity capable of enforcing the degree of social order necessary for the performance of a given activity or event. In this formulation, the organizational essence of any OSS corporation is an institutional entity capable of enforcing the degree of social order necessary for the joint production of computer software among status equal volunteers with weak social ties. To mistakenly identify the open source licensing scheme or the

Anglo-American customs of land tenure (or some combination of the two) as the essence of OSS is therefore to set this dissertation's argument in the wrong way.

Aim, Thesis and Procedure

The aim of this chapter is to explain the social institution that functions as the organizational essence of OSS from extant knowledge on the topic. This way of theorization enables us to examine, in the remainder of the data-based chapters, the communicative constitution of the OSS organization of production and the ways in which it articulates and actualizes the economic rationality of the developers who use it.

The forthcoming analysis shows that the Anglo-American institution of land tenure is the sole essence of OSS as it enables a distinct technosocial system of product development. This technosocial system, which came to be known among participants in the longstanding Python OSS project as "Benevolent Dictatorship for Life" or "BDFL," has the de facto authority to coordinate the activities of volunteers in OSS production and thereby to overcome technosocial problems of coordination and scaling.

To adequately construct this finding, the analysis first shows that the legal essence of OSS as a kind of software product cannot, and was never meant to, enforce any degree of organizational order upon the production processes through which it is created. The analysis then examines the BDFL institution in

context-general terms to show that this political institution is the essence of OSS.

In so doing, the analysis shifts attention from the immediate case of PKW eKnights to the larger case of OSS projects due to the fact that participants in long established enterprises such as the Linux or Python projects had already contended with similar technosocial problems of coordination and scaling (Weber 2004). Indeed, some of PKW founders were associated with the Python OSS community of practice and so had prior understandings of such problems and their methods of solution and were able to then adapt these methods in the Israeli scene of open government data by way of mimetic isomorphism (DiMaggio and Powell 1983). This isomorphic process of adaptation and its importance to PKW organization of production is elaborated in chapter 5.

The Legal Essence of OSS

In order to fully understand why the legal essence of OSS as a software product cannot solve technosocial problems of coordination and scaling among members of an OSS corporation, one must grasp three fundamental properties of computer software.

First, any piece of software consists entirely of a list of instructions known as "source code." Second, software developers write source code in "programming languages" or notation systems that humans can read and understand as well as fix and modify (Weber 2004:4). Third, the writing of any source code occurs within the context of a vertical hierarchy of programming

languages with decreasing degrees of readability, which goes all the way down to the machinic circuitries of a computer hardware:[39]

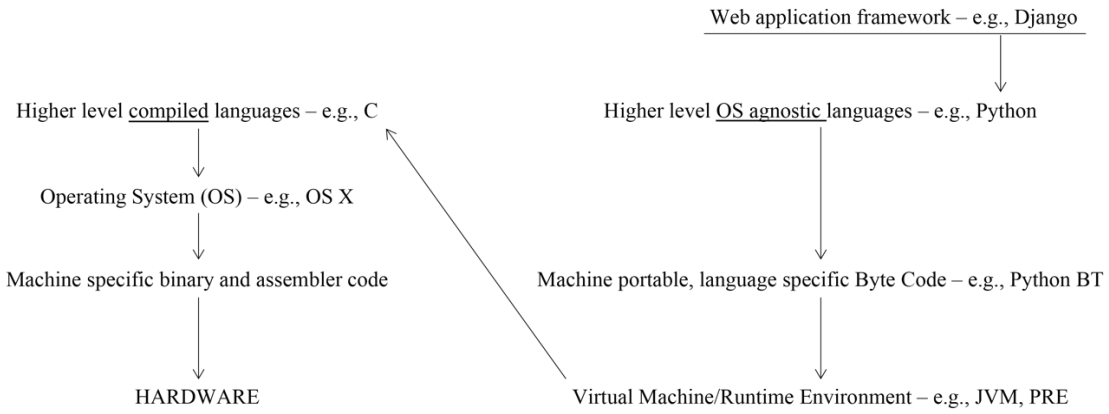


Fig. 9. Hierarchy of Computer Code in Web Software Development

Though the following description is somewhat technical, it remains necessary for a comprehensive view of the conditions for the possibility of software development, especially web development, and, more importantly, it will assist in our understanding of the legal explanation to come.

As apparent from Fig. 9, at each level of the hierarchy of computer code, a translation occurs between adjacent languages or levels of code. Instructions from the code or language at a higher level must be translated into a form whereby they can be read and executed by the lower level code with minimal loss of meaning (Born 1997:145). The system is thus comprised of a hierarchy of mediations, the lowest level of which is known as

[39] This model was adapted from the work of media anthropologist Georgina Born (1995; 1997). Given that computer science and engineering underwent significant changes and developments since the publication of this early work, I had to revise and update Born's original model. This secondary work was conducted under the purview of the two native experts mentioned in chapter 2.

machine code — the instructions that drive the hardware. These instructions are written in binary form (i.e., a long string of ones and zeros) that a computer can read and execute but a human cannot read (Weber 2004:4). The next level is assembler code, made of mnemonic abbreviations of binary code. These two basic levels in the hierarchy of codes can be considered singly, as the assembler code maintains a one-to-one relationship with the binary code and the hardware it represents.

At a higher level we find the general Operating System (e.g., Apple's OS X, Microsoft's NT Windows or any free distribution of UNIX or Linux), which is responsible for access to and management of the whole system (including all network communications), and which provides a framework for higher level programming in compiled languages (e.g., C or C++). Within the hierarchy of codes, this level is the basis of modern computing as it provides programmers with more intuitive notation systems that condense ways of expressing many thousands of lower-level operations that run via binary/assembler code.

A defining feature of high-level languages is that the lists of instructions written within them must be "compiled" or made into binary/assembler code in order to be read and executed by the machine. One limitation of such "compiled languages" is, therefore, their dependency on all the levels of code beneath them. The paradigmatic way to overcome this limitation in contemporary computer science and engineering has been to create an abstract model of a computer system where hardware and OS

dependencies are replaced by a requirement for language specificity. At the base of such systems there is usually one of two types of computer programs: Virtual Machines and Runtime Environments. The central function of these programs is to emulate the execution of lower level instructions. Byte code, the notation system in which these executable instructions are written is similar to assembler code with one important difference. Whereas assembler code is tied to the material hardware on which it runs and operates, byte code is tied only to the higher-level instructions it encodes. Assembler code is therefore specific to the machine below it by material necessity, whereas byte code is specific to the language above it by conceptual convenience.

The languages populating this highest level of the hierarchy of codes are generally more intuitive and less complex than their compiled counterparts. Popular examples of such languages, which I propose as *agnostic languages*, are Java, Python and Ruby.[40] Agnostic languages, which were originally designed to be general-purpose languages, provide developers of websites and online applications with means to create 'tool boxes' for their specializing purposes. These tool-boxes or Web Application Frameworks consist of sets of components, such as readymade code formulas for the writing of specific functionalities, which can be easily applied in the construction

[40] I thank Shiri Dori-Hacohen for this conceptualization.

of particular kinds of websites. For example, the Python web application framework known as Django was devised especially to support the development of complex database-driven websites. While the Django framework is built on top of the Python programming interface, it does not truly exist above the level of agnostic languages in the hierarchy of codes. That is, Django code is a specialized application of Python and the programming interface in which it is executed can therefore be considered as a Django/Python interface.

A source code written in an agnostic language such as Django/Python — which happens to be the standard programming language in PKW volunteer setting — translates into lower level instructions in byte code. These instructions are both machine portable and language specific as they are constrained only by the notation systems of which they are part. Then lower-level instructions in byte code are read and translated into machine specific assembler or binary code by the compiled language in which the Virtual Machine or Runtime Environment that emulates their execution was written (usually C or C++).

From here we see that whereas the level of byte code in the hierarchy of codes is both machine and OS portable, the virtual machine or runtime environment program on which it depends for its execution is not. In other words, while the compilation of higher-level agnostic code into byte code is context-general, the compilation of byte code into binary/assembler code is context-specific. Much of the contemporary software development is web

software development, and much of that web software development is done in agnostic languages. And so, while compiled languages remain central for the operation of computer systems and for the development of software that requires particularly high degrees of machine compatibility, agnostic languages predominate the contemporary scene of web software development in which PKW volunteer setting is situated. Maintaining the distinction between compiled and agnostic languages is important because it allows a better understanding of the legal principles behind proprietary and Open Source Software (OSS), to which I turn next.

What is Proprietary Software?

Software production within a compiled language such as C++ usually ends with the objectification or reification of the original source code into an artifactual version of binary/assembler code. Such code artifacts are usually incorporated into software packages as executable files. Commercial software has been traditionally released in this form.

Given that a software's source code is ultimately a recipe for the binaries that run the actual program on one's hardware, developers who have the source code can understand what the original author was trying to accomplish when he developed the software. The implication is that these other developers can easily fix and modify the software as they see fit. By contrast, if the developers only have the binaries, they can neither understand nor modify the source code. Shipping binary machine

code is therefore a very effective way for commercial firms to control what programmers can do with the software they buy (Weber 2004:4). While such traditional ways to ship computer software are still very much with us, online websites — which are the dominant type of contemporary web software — cannot be reified and shipped in this way.

A website is a client-server software that runs on both the personal computers of internet users (the "client side") and the remote server machines that deliver information to these users (the "server side"). In contrast with the conventional software package, a website's agnostic source code is executed into machine level operations on demand. Owners of commercial websites therefore need to use other methods if they are to hide their source codes from the prying eyes of internet users — a practice accepted among the majority of commercial firms, although value in the internet rarely comes from a website's source code alone.

In general, it can be observed that computer users have access to some of the client-side code that their systems execute, while company employees and other authorized parties have access to the level of agnostic languages in the hierarchy of codes operative on the remote server. Access to server machines from outside a company's network is also ordinarily restricted by authentication procedures. Additionally, programmers can use unintuitive naming conventions for variables, functions and other code entities so as to make the client-side source code of their websites less transparent to computer

users.[41] Whatever the procedure for obscuring higher level code, the bottom line remains the same; the logic of private property and ownership prevails — that is, the right to exclude consumers from that which one has successfully laid claim to.

What is Open Source Software?

Open Source Software (OSS) inverts this taken-for-granted notion of private property and ownership. The essence of OSS as a product is not merely that software is available for use, free of

[41] An important caveat in this regard is the growing tendency of commercial firms to open some of their source code such that external programmers could use it to develop additional features for the original software. The origin of this approach can be traced to the famous "Browser Wars" between the Netscape and the Microsoft corporations. In 1997, Netscape, a popular browser vendor, was battling severe financial losses due to a competition posed by Microsoft's Internet Explorer browser (Coleman 2013:78). In attempt to remain in business, Netscape's executives announced in 1998 that they would release the source code for their browser under an open source license, and invited the influential OSS developer, Eric Raymond, and several of his peers, to a meeting to help them plan their new strategy (Streeter 2011:159). Those in attendance at the meeting saw this as an important opportunity to get the commercial corporate community to take free software seriously, and towards that end chose to follow a pragmatic path of using the term "Open" (instead of Richard Stallman's term "Free") and of emphasizing technical advantages rather than normative ideals (ibid). Soon after, an organization called Open Source Initiative was founded to support these efforts. As opposed to the communitarian idealism of Stallman's Free Software Foundation, Raymond who wanted to develop sustainable business models for OSS, argued that closed source software is economically less efficient, and that large corporations that monopolize software through copyright laws hold innovation back as they prevent free competition in the marketplace. Raymond's evangelism has proved effective; today, corporations spend millions of dollars developing and advertising open source software. From its inception, the Open Source Initiative's challenge has been to generate licenses that balance the requirement for companies to commodify software outputs with the increased potential for productivity, made possible by involving outsiders and harnessing their contributions (Tkacz 2012:392).

charge, but more importantly that it provides the instructions for a given program along with the binaries to anyone who is capable of and has interest in using and/or modifying it. Free source code is open, public, and non-proprietary. As such, it includes the right to run the software for any purpose, to study how it works and adapt it to one's needs, and to improve the software and share one's improvements with other programmers who use it (Weber 2004:5).

The source codes that OSS developers share are protected by copyright licenses that typically fall into one of two categories: Berkeley Software Distribution (BSD), or General Public License (GPL). BSD-style licenses are short documents that specify three basic provisions. First, users have explicit rights to unlimited distribution of the software in any form of code with or without modification. This means that users are allowed to develop proprietary software out of the source code, and are not required to pay royalties to the original authors. Second, users must give credit to the authors of the source code by retaining and reproducing the developers' copyright notice in the derivative products. Finally, the BSD protects the developers from any legal liability that might be associated with the usage of their software in any setting (Weber 2004:181).

In contrast with the permissive attitude of BSD, GPL-style licenses are detailed, constraining and explicit about the ideological principles behind them. The GPL includes a substantial preamble that explains the underlying principles of

the license and why it is constructed the way it is (ibid:182). It implements this statement about what the world of software should be by requiring that whoever distributes the software must also distribute the source code and any modifications to that code. Further, users are not allowed to combine the software with any proprietary software unless the entire combination is then released as free software under the GPL. This last rule constitutes the so-called "viral" character of OSS.

The difference between the two types of OSS license derives from the disparate sociohistorical contexts in which they were generated. The permissive attitude of BSD reflects its heritage as a byproduct of university research, particularly during the 1970's when American computer scientists were thinking about ways to connect between university-based research and the larger technological community, including commercial firms (Weber 2004:181). The license thus serves to credit the developers, protects them and their institutions from liability, and then lets programmers do what they want with the product (ibid).

By contrast, the restrictive attitude of GPL reflects the communitarian ideology of its maker, Richard Stallman, who in 1984 created the Free Software Foundation as a countermeasure against commercial attempts to take control over all software through the enforcement of traditional copyright laws.[42]

[42] For a more complete account of this history, see Weber (2004), Kelty (2008), Streeter (2011) and Coleman (2013), to name but a few of the resources I have used.

To date, PKW civic websites are published under a BSD-style license. The reason for this is not so much ideological as it is practical. The restrictions of Stallman's GPL are such that the participants find it difficult (if not impossible) to avoid using features that cannot conform to the last and most significant rule of this license.

Can the Legal Essence of OSS Solve Technosocial Problems of Coordination and Scaling?

From a microeconomic perspective, it is easy to see that PKW participants' choice of publishing their civic websites under an OSS license is partly a means to attract contributions and thereby to decrease the overall time of production. However, it is not at all clear that such an economic interest can, itself, determine the corporate structure of the eKnights. In fact, it seems that the only OSS process that can be explained in strictly legal terms is that of "forking," a concept that describes a procedure of division rather than unification and cooperation. By legal definition, any participant in an OSS project may "fork" or create a copy of his project's source code and start an independent project from there. While forking may be used as a reaction to antisocial behavior among members of a given project, it has no inherent value with respect to sequencing or ordering social action. In the case of a successful fork, participants in the new project will encounter the same technosocial problems of coordination and scaling encountered by their counterparts in the original project.

We may therefore infer that the organization of OSS production cannot be based on the legal essence of its product. This finding, taken together with the observation that the shared economic interest of OSS developers to create an efficient organization of production cannot determine the corporate structure of such organizations, leads us to accept that the institutional essence of OSS must be political.

The Political Essence of OSS

The political essence of OSS has to do with the character of a "project founder" or the developer who conceives the product and who customarily writes the initial and most essential lines of its source code. Once such a developer chooses to publish a piece of source code within the framework of a new voluntary corporation, other developers with similar interests may find it more economical to join that original effort than to fork the project's source code (i.e., develop a version of it on their own or within the framework of a different project).

In any such event of cooperation, all parties will accept without question that the original author has an ultimate right to exclude others from his production process. That is, OSS project founders get to decide which contributions will enter the source codes of their products and which contributions will not. This institution of gatekeeping, which came to be known among OSS Python developers by the term "Benevolent Dictatorship For Life" or "BDFL," provides an effective solution to technosocial issues

of coordination and scaling. Insofar as project founders control access to, and management of, their projects' source codes, they can force other contributors to submit (rather than implement) their additions and modifications.

BDFL thus comes to the fore as one — and I would argue the best if not the only — institutional entity that can enforce a degree of social order upon corporate members who operate under the condition of lack of command ability. A benevolent dictator may allow everyone to fork his source code while not being able to tell anybody what to do. However, in so doing, he gains full control over the production of that particular version of the software. In this way, the traditional notion of private property is reproduced in its original form.

Cultural Origins

The OSS notion of BDFL is best understood within the Anglo-American discourse of land tenure.[43] Specifically, it resonates with John Lock's typology of claims to land ownership in The Second Treatise of Government (1689). In Locke's formulation, seekers of land are first encouraged to create homesteads on unoccupied frontier territories. Second, Lock permits the acquisition of a land title from a previous homestead owner who may or may not be present but still retains a legal claim on the

[43] This original argument was first presented by OSS developer and amateur ethnographer Eric Raymond ([1999]2001), and was further developed by political economist Steven Weber (2004).

land. Finally, in the occasion where a land title is lost or abandoned entirely, Lock permits for the claiming of the territory through adverse possession by moving onto the land and improving it.

Observers have noted that the first and most pervasive OSS custom of project ownership is the act of initiation, i.e., the act of publishing an original source code for a new software. This aligns closely with the early American practice of frontier land claiming for agricultural development, a system made official by the U.S. Federal Homestead Act of 1862. Second, it is customary among OSS developers to publicly observe the passing on of a project to an identified recipient. In this custom, the owner not only has the right but also the responsibility to hand down the project to a competent successor when he is no longer interested in occupying the position of BDFL (Weber 2004:162). Finally, OSS developers may lay claims to project ownership by demonstrating publicly that they have attempted to pick up an abandoned project whose owner has disappeared from the scene. A developer who attempts to revive a project in this way is expected to make substantial efforts to find the original owner and wait a reasonable period of time for reaction from the community to any proposal to take over the project. It is also customary that ownership acquired in this way is not fully recognized until the new owner has made substantial changes to the original source code and brought them out into the open.

Obviously, this custom has much in common with Locke's sense of claiming a land title through adverse possession.

Raymond ([1999]2001) further proposes to think of the 'frontier land' spanned by OSS initiatives as the *software noosphere* or the ideational space of all programming thoughts. This analogy brings into view the dual nature of the source code that inaugurates the beginning of a new OSS project as a material and symbolic 'landmark.' Such 'landmarks' are material because they settle the technical conditions and parameters for software production, and symbolic, because they delineate the pieces of 'programming thought' on which software are built and to which the founders have laid claims.

Guiding Metaphor

To further our understanding of Raymond's land ownership metaphor, we can think of source code for a given product as an 'agricultural farm;' the typical OSS product as a kind of 'produce' cultivated on that farm; and the typical project founder as both the 'chief engineer' of the farm and the owner of the piece of land on which the farm is conceived and built.

In the usual scenario, an OSS project's source code qua 'agricultural farm' has no a priori design or blueprint. A few of the workers, the principal developers of this farm, will construct complex buildings, while many others will be occupied adjusting and repairing subordinate components of the owners' more ambitious creations. Under the condition of lack of command

ability, all individual workers have rights of voluntary participation and voluntary selection of tasks, which means that they may build the farm as they see fit or abandon the project/farm anytime they choose. At the same time, project founders qua BDFLs preserve the right to decide if and how these modifications will be integrated into their 'farms.'

This elaborated metaphor helps us to see how project founders are able to distribute or publish their products under OSS licenses while obtaining traditional ownership over the creation of such products. The fact that H has a legal right to fork and develop S's source code does not in any way mean that H has a social mandate to access and/or manage S's source code or that S has an obligation to accept H's contributions.

Conclusion

The analysis developed in this chapter shows that the institutional entity capable of enforcing the degree of organizational order necessary for the resolution of technosocial problems of coordination and scaling in OSS corporations is BDFL. This institutional entity, which resonates naturally with participants in most OSS projects, derives its efficacy from the traditional, Anglo-American discourse of proprietary laws. That is, BDFL affects the actions of volunteers in OSS corporations by

means of exclusion and gatekeeping that have much in common with the assumptions behind proprietary software.[44]

While some large-scale projects[45] have discarded the benevolent dictator model at some evolutionary point,[46] Raymond ([1999]2001:103) finds that more complex governance structures are considered by the participants to be less stable.[47]

The most salient example of a large-scale OSS initiative in which the BDFL institution has remained particularly strong is van Rossum's Python project. From OSS folklore, it is widely understood that the term was first used by members of the Python community who, in 1995, began to refer to van Rossum as a "Benevolent Dictator For Life" or "BDFL." Their usage of this

---

[44] The only difference between the two seems to be the point in the cycle of production where one has social legitimacy and political power to exclude others.

[45] A large-scale project consists of several hundreds of central members who do most of the work, and several thousands of comparatively peripheral participants who contribute in more indirect and sporadic fashion (Weber 2004:71). In my observation, these proportions are maintained in projects that consist of smaller groups of committed participants who are surrounded by larger communities of potential volunteers. It is therefore reasonable to assume that in the absence of such communal context, core developers must make special efforts to publicize their projects if they are to elicit peripheral contributions for their software products.

[46] One example is the Apache project in which the co-developers constituted a voting committee. Another is the Perl project in which the co-developers operated a system of rotating dictatorship whereby control was occasionally passed from one member to another (Raymond [1999]2001:101).

[47] Beyond the apparent fact that the work of complex committees requires time and effort, Raymond ([1999]2001:103) proposes that it is hard to fit into the Lockean model that the participants use for reasoning about the simpler cases. In projects with more complex structures of governance, it is harder to do an accounting of ownership, and thus to avoid conflict unless the group enjoys an exceptionally high level of harmony and trust.

term reflects a consensual agreement that van Rossum has the final authority to make decisions and resolve conflicts with regard to the Python language source code.

From the standpoint of CuDA, the finding presented in this chapter can be regarded as a summation of the context-general, technological and political conditions, which must be presumed – implicitly or explicitly – so that a given OSS corporation's discourse of work could be what it is. In the interest of theoretical precision, this summation can be best extracted as the following line of reasoning:

- The macro-social transition from traditional to modern society creates a structural opposition between work and leisure.
- This structural opposition establishes the condition of lack of command ability – i.e., the inability of participants in voluntary corporations to justify or legitimate positions of authority.
- This condition of lack of command ability primes the creation of a basic contract with two reciprocal norms: voluntary participation and voluntary selection of tasks.
- This social contract together with the cultural givens of land tenure acquisition in the Anglophone West lead to the creation of the BDFL institution. That is, adherence to the norms of voluntary participation and voluntary selection of tasks creates a political-economic situation where BDFL is the most efficient regime-type capable of enforcing social order upon participants in OSS production.

CHAPTER 5

THE LOCALITY OF BDFL

Introduction

The previous chapter has shown that Benevolent Dictatorship For Life (BDFL) is one, and probably the most efficient institutional entity capable of enforcing the degree of social order necessary for the resolution of technosocial problems of coordination and scaling within voluntary corporations whose members operate under the conditions of time scarcity and lack of command ability within the modern realm of leisure.

The present chapter completes this analysis by showing that PKW founders did, indeed, adapt the BDFL institution, and that this local adaptation was a necessary and sufficient condition for the creation and sustainment of the group's first and most celebrated eKnights. This finding provides a proper context against which the following chapters (6 to 8) interpret the practices of "terumat cod" (code donation) and "la'asot cod" (to do/make cod) as particular manifestations of the general relationship between the explanatory and response variables of economic rationality and social organization.

The present chapter accomplishes its aim in four stages. First, the method by which the analysis establishes its empirical finding is briefly presented and discussed. Second, the historical narrative that accounts for the process by which PKW founders adapted the BDFL institution to their local volunteer

setting is constructed in detail. Third, this process of adaptation is discussed in relation to DiMaggio and Powell's (1983) notion of isomorphism. Finally, the way in which that isomorphism contributes to the analysis of "terumat cod" in the following chapter is stated and explained.

Method

The Voluntary Association of The Public Knowledge Workshop consists of two distinct groups, an administrative body and a community of small voluntary corporations. This duality is a product of two complementary processes. First, an official process of institutionalization that led to the creation of PKW as a Voluntary Association whose productivity relies primarily on an unpaid workforce, and second, an unofficial process of structuration that shaped this workforce into a segmented community of eKnights. Neither of these processes was necessary, and each could have taken a different route at several different historical junctures. By connecting some of these junctures, the present chapter aims to show that the endogenous adaptation of BDFL into the Israeli scene of open government data was a necessary and sufficient condition for the creation of PKW regardless of any other exogenous pressure or constraint.

To establish this finding, the forthcoming discussion starts with an examination of the embryonic stage of PKW as a single person's project and ends with a descriptive account of

the OSS customs of project ownership that were active at the time of my fieldwork in 2015.

The task of recounting the history of PKW is challenging not only because there are many ways to tell the story, but also because there is no official record that documents it. PKW history is delivered orally, and many of the founders who witnessed and played crucial parts in it are no longer active in the group. Among these founders, only one person agreed to share his personal narrative within the framework of an in-depth interview. Fortunately, this person is the individual who started the project that evolved into the social scene that I observed when I began my fieldwork in Tel Aviv and Jerusalem. As with any narrative, it is partial and contestable; nevertheless, it does construct a plausible chronology. I am taking for granted that the sequence of events that the narrator recounted is generally accurate and that this person's perspective was relayed in earnest, as I could not find counter evidence of it in any of the study's datasets.

The Creation of PKW

According to all available sources, the story of PKW starts in the winter of 2009 with a young officer (call him Elihav), who was about to complete his military service in the science and technology unit of the Israeli army, when he found himself unable to make an informed decision about who to vote for in the upcoming elections to the Knesset (Israeli parliament). This was

due to a lack of information about the parliamentary conduct of the competing candidates.

Being a skillful data analyst, Elihav decided to search the official Knesset website for legislative information that might disclose additional information about parliament members and their political records. What he discovered was that, while the website contained detailed archives of bill information that tracked back to the first Knesset in 1951, it lacked the functionality necessary to generate analyses of its own raw data.[48] This moment represents a crucial turning point in Elihav's narrative as many citizens would have reasoned that any attempt to make sense of such a technologically limited archive would be not only uneconomic but likely impossible.

Elihav, however, who had some experience using the Python programming language decided to "sit and write" a "web-scraping" device capable of extracting and reconstructing elements of the Knesset database into an analyzable spreadsheet. To put it in more formal terms, Elihav's utility of gaining civic knowledge from the Knesset's website outweighed his disutility of spending his scarce leisure time in the performance of difficult work.

However, once Elihav undertook his analysis, he found the task more complicated than he had initially anticipated. There

[48] For example, in order to figure out the voting patterns of individual parliament members on specific kinds of bills (such as bills on environmental issues), one had to collect all the relevant data manually by looking at the web-page of every single vote concerning each individual bill.

were multiple ways to organize the data and a growing number of statistical variables and operations that had to be taken into account. The analysis not only required a significant investment of leisure time, but also a considerable understanding of the content itself.[49] From these obstacles, Elihav made two consequential realizations relative to the creation of PKW. First, Elihav understood that in order to carry out his task he would need to find the means to collaborate with other interested parties. Second, it became apparent that he had uncovered a trove of information with immense value to the Israeli polity. Elihav thus concluded that it would be beneficial to attract people with both similar civic interests and a willingness to help. In his words, "…at first it came from an egoistic place; I understood that I can get the data but have no idea what to do with it."

The Creation of Open Knesset

Encouraged by enthusiastic reactions from family members and friends who acknowledged the significance of his initiative, Elihav embarked upon a search for potential collaborators, leading him to an activist group called The Movement for Direct Democracy" whose members had a particular interest in intervening into the parliamentary system of the Israeli Knesset.

[49] For example, Elihav had difficulty performing the basic task of classifying bills into distinguished topics that he considered as an initial step towards analyzing the relationship between individual parliament members and agenda categories.

After some initial interactions, Elihav found that the members of this group were much more inclined to "discuss issues thoroughly" than to "get things done." Beyond the fact that these activists lacked the technical skills necessary for effective engagement with the data, their deliberative approach decreased Elihav's available leisure time to actually work and develop his project. In Elihav's economy of speech and action, speculative talk about abstract issues under the condition of time scarcity was an impediment on the completion of programming tasks. To him, useful conversations were ones in which people devised practical solutions to technical problems.

This hands-on approach is now a definitive feature of PKW discourse of work, where the participants who "do" distinguish themselves from those who only "make noise." This opposition is thoroughly analyzed in chapter 7. For now, it is important to recognize that while members of The Movement For Direct Democracy did not know how to code, and principally preferred to "discuss things thoroughly," they may simply have lacked the ability and knowledge to create an efficient organization of production under the condition of lack of command ability.

While Elihav did not make this inference explicitly, his intuition led him to find people who did have such knowledge and skills. These individuals were located primarily through a professional forum of Israeli Django/Python web developers called PyWeb-IL, generally populated by engineers capable of assisting him with his technical tasks. PyWeb-IL participants who used to

hold periodic meetings and networking events invited Elihav to introduce his ideas at one of their gatherings. Elihav took this opportunity to present his story with an emphasis on the programming procedures that he used and the difficulties he had encountered. Following this presentation, some audience members who showed civic interest in the Knesset website's database agreed to assist Elihav on the condition that this initiative would be an OSS project whose source code would be uploaded to the GitHub platform for software production.[50]

Elihav accepted the proposal and soon after began working with one of the participants attending his presentation, a developer who volunteered to assist him in his programming tasks (call him Ilan). The original project now included an attempt to construct a Python based website that would render much of the Knesset database accessible to Israeli internet users (and any other interested online public for that matter). Ilan, who was a veteran in the Python OSS community, the founder of a profitable startup company, and a well-known figure in the Israeli high-tech scene, committed himself fully to Elihav's project, bringing his own vast programming experience and cultural competence as an OSS developer to the creation of the new website.

From 2009 to 2010, Elihav and Ilan worked together with only occasional assistance from other participants in the PyWeb-IL forum. By mid-2010, the first iteration of the website's

[50] To be sure, this was a choice of convenience as much as it was an ideological preference.

source code reached the point where it was ready for basic use. In order to expand the range of distribution and attract new volunteers to the production process, the two developers purchased a dedicated domain from an internet service provider and published the website under the name "Open Knesset." Around the same time, they also established a meeting schedule to encourage new volunteers to join the project locally through face-to-face engagements, which gave rise to their weekly development meetings, the social situation that remains the basis of PKW participatory model.

The development meetings successfully attracted new participants, and by August 2010, Elihav and Ilan gathered a group of volunteers for an intense development effort within the framework of a weekend hackathon. Following this event, a regular team of about ten Python developers continued to participate at the group's development meetings and the project grew steadily from there. Elihav was now publicly recognized as the undisputed owner of Open Knesset. Because the OSS developers who participated in Open Knesset were Django/Python developers, and since the term Benevolent Dictator For Life or BDFL originated in the Python project, Elihav was recognized by that title. This is made apparent in the following excerpt where Moti, a veteran PKW participant, shares a version of Open Knesset's creation story to students in Hackita02 program:

(2) Hackita02 (2/12/2015)

1. How a project starts? Like come on and think about it. Like-maybe this way we will also do some survey of open knesset.

2. But there was Elihav, yes? Elihav, he built some scripts to do scraping on the knesset website, to derive statistics for the upcoming elections. This was five six years ago. […] So he wrote some scripts of that sort, and so it grew and another developer, Ilan, joined him.
3. But eventually the project's [owner] is Elihav. Like he is the project's father, he is the leader. It's called bi di ef el. Benevolent ah dic- dictator for life. It's a dictator for life. So that how this guy Elihav is.

The Segmentation of Open Knesset

On 2 December 2010, the deadliest forest fire in Israel's history broke out on Mount Carmel near the northern city of Haifa. The fire claimed forty-four lives and hundreds of thousands of acres of forest. A failure by The Israel Fire and Rescue Services to contain the blaze led the Israeli government to request international assistance. While providing this assistance, various officials in European countries expressed surprise at Israel's helplessness. Many asked how it was possible that a country whose defense budget was estimated at dozens of millions of dollars at the time could have had such a significant shortage of planes and firefighting materials. Soon after, the Israeli press started to interrogate the conduct of government officials in the Ministry of Finance, who, in turn, broke into a dance of mutual accusations. The Israeli public wanted answers that no official was able or willing to provide.

Following the hackathon that Elihav and Ilan organized around the time when the public controversy about the Carmel forest fire was at its height, one of Open Knesset's participants (call him Ilya) came to the conclusion that the failure of the

Ministry of Finance, and the identities of government officials who should be held responsible, could be discovered through available government data that documented decisions of the Finance Committees related to the forest fire controversy.

Like Elihav before him, Ilya investigated the official State Budget website to discover that the data was provided in at least four incompatible formats that were not amenable to systematic analysis. Ilya brought these results to the attention of the members of Open Knesset's inner circle who agreed that "if the Knesset is the philosophy of state administration, then the budget is the practice," and that, therefore, exploring the movements of funds by the Ministry of Finance would be just as important as studying legislative processes at parliament. With this encouragement, Ilya decided to start the development of a second civic website called Open Budget that would deal specifically with the state budget data.

This development is significant to the current discussion for the following four reasons. First, the creation of Open Budget provides concrete evidence that PKW founders followed the first OSS custom of project ownership, which is to physically initiate the project with written code, not simply with an idea. To put it in Elihav's words:

(3) Interview (1/30/2017)

1. Similarly to the way in which I was the beginning of open knesset, there are the first several thousands of lines of code that someone has to write alone.
2. So Ilya sat at home and did that alone, and then came back to open knesset meetings and connected to Meron […] And similarly to

> the way in which I and Ilan were the core of open knesset, they were the core of open budget.

In this commentary, Elihav touches upon the dual nature of an OSS project's first lines of source code as material and symbolic 'landmarks.' By sitting and writing the basis for the Open Budget website, Ilya established the technical conditions and parameters for product development. In so doing, he also laid claim to the territorial piece of creative programming/civic thought on which that product was to be developed and made.

Secondly, Ilya's creation of Open Budget provides an example of a localized variation of the evolutionary path of most documented OSS communities of practice. On Raymond's ([1999]2001:101-102) account, projects that start with a single developer or an owner/maintainer tend to mutate into small partnerships between an owner/maintainer and one or two committed co-maintainers who usually have privileges of source code management. In the next evolutionary stage, the owner/maintainer starts to function as a BDFL while the project grows with less committed or occasional code donors now also offering contributions to the project's source code. Such BDFL organizations can grow infinitely as far as we can tell, and when they do grow, they tend to construct a pyramidal form of gate-keepers as the owner/maintainer shares more of his direct code management authority with the committed participants who Raymond calls co-developers and who often appear to be the projects' original co-maintainers. Indeed, the majority of the largest and most successful OSS projects are of this kind. Against this

backdrop, it is apparent that when Ilya chose to start a new project, he in effect initiated a significant transition from the evolutionary path of the typical OSS project.

This transition speaks directly to a third point about the creation of Open Budget that bears special significance to the current discussion, i.e., the explanatory advantage of microeconomic theory over forms of sociocultural explanation. From a traditional cultural perspective, one could hypothesize that PKW participants' transition from a single to a two-project community can be best explained by their particular values and beliefs — for example, their local perceptions of the relationship between the individual and the community. While such explanations are generally valid and important, they could miss a crucial element when it comes to economic activities of production and exchange. Indeed, it is clear that a central reason for the creation of the development meetings, themselves a deviation from the online nature of most OSS communities of practice, and the choice of participants to create two different projects with different civic purposes and different technical infrastructures, was the basic need for a volunteer workforce. In Elihav's words:

(4) Interview (1/30/2017)

1. We worked together, you know, on two sides of the same table, and when a new volunteer arrived, we showed him both projects [so he could pick one] according to what he was more interested in, or according to the technological stack that was closer to him, because there was a little technological difference.
2. This is the best example of why the management of projects in the open source world is very different from the management of projects in a commercial firm or any other entity. [The

> difference is that] you are not operating in a closed economy. By the fact that we started another project- like every startup will tell you no way, this is stupid. You cannot at this stage of a project's life start another effort of product development […] I think it was a great move because it significantly increased the amount of people we could bring. [For example,] someone [could] say oh okay open knesset is interesting, but wait a minute, maybe open budget is [more] neat? Or one [could] hear about the open budget project [and then realize that he] is not able to contribute to it because he doesn't know the technological stack, and [then] find his way to open knesset.

While this economic rationale may explain why PKW founders decided to run two parallel OSS projects, one may still wonder why previous OSS projects — at least the most successful and known ones — did not follow a similar evolutionary path. I believe that a partial explanation has to do with the relationship between the locality of the products and the centrality of Tel Aviv as a cultural and industrial center in Israel. Elihav and Ilan as well as other participants in the PyWeb-IL forum — also based in Tel Aviv — were in close geographic proximity to one another and to this urban center. Given the significance of Tel Aviv to the Israeli high-tech scene, I would even suggest that if one were to start an online OSS project that focuses on local civic matters with the aim of attracting developers from disparate cities in Israel, that project would most probably fail.

Finally, and most significantly, the creation of Open Budget provides direct evidence for the institutional nature of BDFL on two interrelated grounds. First, the fact that Ilya could have replicated the organizational order of Open Knesset retroactively means that the same order could have been replicated from the original Python OSS project. Secondly, the

ability of different individuals to claim ownership over the management of their projects' source codes in an approximately similar way refutes explanations of project leadership that focus on the idiosyncratic personalities or characters of individual project founders (although these psychological factors certainly enter the OSS discourse of leadership).

The Institutionalization of PKW

At the beginning of this analytical section, I have argued that the evolution of PKW involved two simultaneous and to some extent contradictory processes: an unofficial process of structuration that led to the creation of PKW as a segmented community of voluntary corporations, and an official process of institutionalization that led to the creation of PKW as a Voluntary Association. The analysis so far has followed the first of these processes by describing how Open Knesset emerged and then split into two similar voluntary corporations in an organic, endogenous fashion.

It is now left to show that PKW founders sought an official process of institutionalization for reasons other than a deficient BDFL institution.[51] Indeed, we find that aside from

[51] It should be noted that the participants faced a real-world scarcity of developers as both projects were also relatively ambitious and thus resource demanding. In this respect, the evidence presented in this study does not suggest that PKW could have kept developing uninterrupted through a constant process of growth and segmentation. The analytical claim is rather that the group could and did rely on the BDFL institution for its basic operation.

basic needs that had to do with the projects' growth (such as the need for publicity), Elihav, Ilan and their collaborators found themselves in a situation where their lack of affiliation to a recognizable organizational entity prevented them from interacting with government and public officials. In Elihav's words:

(5) Interview (1/30/2017)

When we arrived at the Knesset, we realized that we could not continue to conduct ourselves- I mean, they asked us who are you? We told them we are Elihav and Ilan, we are two guys who do things, we want to help. They didn't know how to digest this. […] In their tender template it says you need fifteen years- that in order to apply for the tender you need- you must be a registered body for fifteen years or something like that- I don't know- like they don't know how to work with people. So we realized that we need to institutionalize in some official way.

After considering several different options, including for-profit forms of organization, the core participants of Open Knesset and Open Budget decided to register their initiatives as a Voluntary Association. By Israeli law, any such form of organization must institute a number of entities that include, among other things, a General Meeting, a Board of Directors, and an Audit Committee. In the case of PKW volunteer setting, this administrative body is perceived as a "startup incubator" whose role is to provide the different project teams with legal, logistical and public relation services.

As a small community of practice whose majority of participants are much more interested in software development than in institutional politics, PKW hires a Chief Executive Officer (CEO) who performs most of the necessary administrative

work, and one community coordinator whose job is to assist the CEO and support the different project teams.

In practice, these participants provide the group's eKnights with an administrative shell by representing the group in various institutional and social settings, establishing relations with other civic and public actors, recruiting funds, attracting new volunteers, taking legal measures against state agencies that prevent the publication of government data, and, most significantly, providing the project teams with a regular meeting place where they can work together and interact with newcomers who show interest in their initiatives.

However, these administrators have no right or power to intervene in the management of the different project teams. In the following commentary, Yona who served as a CEO for PKW gives a clear sense for the independent nature of the eKnights qua voluntary corporations, and for the crucial role that BDFL plays in the internal organization of those teams:

(6) Interview (4/1/2016)

1. Nim: Every table is usually a project?
2. Yon: Yes, they also tend to sit at the same tables […] And they work. They work on their tasks. And in order for a project to succeed/exist, it means that the [BDFLs], the central people in the projects, need to be there. It is a group that sits and works […] in and of itself. It has nothing to do with me. […] The [BDFLs] send an email on Monday noon that says who is coming today? Which is amazing. It's not me doing that. It's the projects' leaders. And it took me time to let go! And to understand that it is also not my job to begin with.
3. Nim: So is [PKW] like a confederation of a lot of small projects?

4. Yon: Just like that. It's more of a startup incubator if you like, yes? To use [an analogy proper] to our volunteers who come from the [high-tech] world.

From this commentary, we see that the BDFL institution is considered as a necessary and sufficient condition for any eKnight to exist in PKW volunteer setting. This, of course, does not mean that the benefit from external institutional and financial support is marginal in any way. The primary point is to show that such administrative support is not necessary nor sufficient for the OSS organization of production within the group's different eKnights.

Project Ownership in PKW

The institutionalization of PKW had both ideological and material implications. On an ideological level, the group became recognized with the growing open government data movement that Michael Schudson (2015) calls "the cultural right to know" in the contemporary civic climate. On a material level, the ability to enjoy the benefits — especially financial benefits — of PKW qua "startup incubator" was understood as a scarce privilege.

These ideological and material constraints add to Eric Raymond's ([1999]2001) geographic imagery the dimension of government transparency and the dimension of administrative support. When a volunteer wants to start an eKnight under the sponsorship of PKW, he must undergo an administrative process that involves the stages of proposal, review and voting. A central requirement for any new initiative at the proposal stage

is to persuade the community how it will advance the organizational mission of making governmental data more transparent and accessible. It is in this sense that PKW ideological orientation toward open government data places constraints on Raymond's 'software production noosphere.' In contrast with other OSS communities, PKW community of practice spans only the 'frontier territory' of Israeli governmental and public data. Any OSS project that deals with civic issues but does not provide a rhetorical response to recurrent exigences of government opacity, inaccessibility and corruption has no place in PKW. Once a new eKnight is accepted to PKW, the Voluntary Association qua "startup incubator" is obliged to provide it with a range of administrative services. Among these services is the allocation of actual territory within the development meetings space. This territory is demarcated by the four sides of a table at which the project teams' members may gather together to work on the individual programming tasks that they self-select.

In the case of PKW volunteer setting, then, the creation of a new eKnight is governed by two societally-transmitted principles. First, a volunteer may claim the position of a project owner or BDFL by 'homesteading' the Israeli 'public data noosphere.' Second, the new initiative must be authorized by PKW administration in order to gain a place within its "startup incubator," and this authorization is spatially marked by the physical placement of a table.

Local Variations of the OSS Ownership Customs

To the best of my knowledge, while PKW participants always follow the first OSS custom of project ownership (i.e., start the project), they are rather ambivalent about the second one (i.e., handing down the project). For example, when Elihav decided to stop leading the Open Knesset project, the institutional position of Yoni, the co-developer who replaced him, remained unclear. On Elihav's account:

(7) Interview (1/30/2017)

1. In terms of leading the project, it was always me. […] But my feeling is that open Knesset has exhausted its purpose. And the purpose was a […] proof of concept that the Knesset data is interesting, that the public wants to consume it without filters, and that the right way to let it consume it is to publish the data [as is].
2. We proved the point. And now all that remains to be done is […] to persuade the Knesset to take ownership of this particular project or another effort that will produce similar results […] And [the] cat and mouse game [whereby] they change the format of something [in their website] and we run to fix our scrapers [so that our website] will go back to work started to bore me.
3. […] At some point, when they did another one of many changes in how their website works […] I said that's it. I'm fed up. Let's replace the open Knesset website with a black page that says sorry friends. If you are upset that open Knesset went down, here is the mail of the Knesset's director general. Send him a letter and tell him you think it's wrong. The [rest of the team] weren't very enthusiastic with [this idea].

In L7:1, Elihav explains that the original purpose of his initiative was to provide a "proof of concept" that the Knesset can and should make itself more transparent and accessible to the Israeli public. On the one hand, the project clearly accomplished this objective by 2015. But on the other, parliament members were unimpressed (or perhaps slightly intimidated) by this result. Given that Elihav's Open Knesset website relied on the database of the official Knesset website for its operation, any change in

that official website required adjustments from the side of his project team (L7:2). At some point during 2015, Elihav decided that he had enough and attempted to persuade his team members to terminate the project (L7:3). These other volunteers were not convinced and kept developing the project's source code in his absence.

In the case of PKW volunteer setting, then, the first ownership custom prevails with the assumption that the person who starts an eKnight has the right to "own it for life" regardless of how much he engages in the development process at any given time. The third OSS ownership custom (i.e., picking up an abandoned project), too, seemed to have little place in PKW, not only because the relative power of the first custom but also because the relatively small size of PKW social scene and the tendency of new project founders to write the source codes of the abandoned projects they had volunteered to adopt from scratch.

Conclusion

The analysis developed in this chapter confirms that PKW founders adapted the BDFL institution into the Israeli scene of open government data, and that this adaptation was a necessary and sufficient condition for the creation of PKW embryonic projects. With this evidence, the present chapter supports the more general claim of Steven Weber (2004:56) that voluntary and commercial corporations should be defined by their organizational features, not by their products. In his words:

> The essence of open source is not the software. It is the process by which software is created. Think of the software itself as an artifact of the production process. And artifacts often are not the appropriate focus of a broader explanation […Thus,] if I was interested in The Secret of Ford, I would focus on the factory assembly line and the organization of production around it, not about the cars Ford produced. Production processes, or ways of making things, are of far more importance than the artifacts produced because they spread more broadly. Toyota, for example, pioneered lean production in a factory that made cars. Twenty years later, this way of making things had spread throughout the industrial economy. Similarly, open source has proved itself as a way of making software.

We thus see that whatever distinctions that may be made between the products of civic and commercial websites are not the principal concerns for PKW OSS projects. It is the BDFL institution, rather than, for example, the democratic and civic principles of open government data, that makes software production within this movement possible.[52] In fact, as chapter 8 discusses, the open government data doctrine could have done nothing to assist PKW participants in their efforts to create a sustainable system of governance for their projects.

To say that the organizational essence of PKW eKnights is the BDFL institution is to confirm the capacity of this institution to spread across geographic and sociocultural boundaries. While voluntary corporations may appear to be very different from each other when one looks at their distinct products and the ideological discourses of their members, they are similar in that they all depend upon the same, predominating, technosocial system of product development.

---

[52] For a thorough comparison between OSS and open government data, see Tkacz (2012).

The Isomorphic Spread of BDFL

The phenomenon of homogeneity that results from the spread of one organizational form among many disparate groups over time is precisely what DiMaggio and Powell (1983) have tried to explain with the concept of *institutional isomorphism*. On their account, there are three types of institutional pressures that make organizations within a macro-societal framework look alike: (i) coercive isomorphism that results from pressures exerted on organizations by other organizations on which they depend for their legitimacy, and by cultural expectations more generally (p.150); (ii) mimetic isomorphism whereby one organization is modeled after another organization in response to conditions of uncertainty (p.152); and (iii) normative isomorphism, whereby the resemblance of different organizations is associated with the professional network that spans them (p.153).

From this theoretical standpoint, the analysis presented in the present and the previous chapters can be read as an attempt to trace a specific path of mimetic isomorphism that starts in the embryonic OSS projects of the 1970's, goes through the original Python OSS project, and ends with the creation of Open Knesset, Open Budget and the rest of PKW eKnights.

Here, the technosocial problems of coordination and scaling demanded a process of mimetic isomorphism not because of ambiguity but because of the distinct condition of lack of command ability that cannot be found in any commercial or domestic corporation. Unable to mimic the production process of

the commercial firm, the pioneers of OSS ended up creating an alternative organization of production. And while an account of the exact isomorphic process that explains how these pioneers adapted the Anglo-American customs of land tenure to the volunteer settings of software production is yet to be written, it is quite certain that once the technosocial system of product development common to these groups has been formalized, and especially within a monologic community of OSS developers such as the Python community, it becomes amenable to mimetic isomorphism. Indeed, it is apparent that contemporary web developers who use the Python programming language as a primary tool for software production emulate the organizational structure of the parent project, and the making of Elihav and Ilan's initiative into an OSS project is no exception. In his inauguration to the scene of OSS production, Elihav learned from Ilan the normative terms that made him the rightful owner of and the final authority over Open Knesset's source code.

Towards an Analysis of PKW Discourse of Work

The overall aim of chapters 4 and 5 has been to establish a proper background against which to interpret the central communication practices by which PKW participants constitute a voluntary organization of OSS production, i.e., the practices of "terumat cod" (code donation) and "la'asot cod" (to do/make code). In establishing this interpretive context, we have learned that the BDFL institution is the organizational essence of OSS.

Chapter 4 has shown, in context-general terms, that the political and normative principle of private property qua the right to exclude others from the production process to which one lays ownership claims is a condition for the existence of any OSS corporation. The reason is the prevailing efficiency of BDFL in overcoming the technosocial problem of coordination and scaling. The present chapter has provided a further confirmation of this finding in context-specific terms by tracing the isomorphism of BDFL in PKW volunteer setting.

An additional goal of this chapter has been to familiarize readers with the particular history and some of the norms of this setting, i.e., the prioritization of the first OSS ownership custom, and the ideological and material constraints imposed upon any participant who starts a new eKnight under the sponsorship of PKW. While these two chapters may offer original contributions to extant research on OSS, I refrain from entering into such a discussion due to the particular needs of this dissertation.

The contextualizing analyses provided in chapters 4 and 5 were conducted only because they are essential to the understanding of PKW practices of "terumat cod" and "la'asot cod" as elements in the group's discourse of work. Similarly, the interpretation of these practices in the following chapters is not an end in itself but rather a means to accomplish the overall aim of this study, i.e., provide empirical evidence that there is a causal relationship between economic rationality and social organization, and that this causal relationship depends on local

communication practices for its actualization. The following chapter takes a first step in this direction by addressing the practice of “terumat cod” as the communicative kernel of PKW organization of production’s technosocial system and the BDFL institution of which it is a part.

CHAPTER 6

ACTIVITY SYSTEM

Introduction

The previous two chapters have shown that participants in OSS voluntary corporations such as PKW eKnights are able to overcome problems of coordination and scaling in technosocial systems of product development by the institution of Benevolent Dictatorship For Life (BDFL). Chapter 4 explains where this institution comes from and why it is likely to emerge organically under conditions of time scarcity and lack of command ability, and chapter 5 provides evidence that PKW founders adapted the BDFL institution to the Israeli scene of open government data by way of mimetic isomorphism, and that this adaptation was necessary and sufficient for the creation and operation of their OSS initiatives.

In terms of the dissertation's overall aim, these contextualizing analyses provide a framework for a detailed description and interpretation of PKW discourse of work and how this discourse enables the causal relationship between economic rationality and social organization. As a first step towards the accomplishment of this aim, the analysis conducted in the present chapter; focuses on the communication event of "terumat cod" (code donation) by which PKW participants constitute the BDFL institution and the technosocial system surrounding it vis-à-vis problems of coordination and scaling. In so doing, the analysis

provides more essential evidence in support of the thesis that the principle of rational action/choice can be used to explain how a given cultural discourse and the organizational forms of which it forms a part came to exist in the first instance.

The finding of this chapter is constructed as follows. First, the ethnographic methods by which CuDA scholars study communication events is presented and discussed. Second, PKW is conceptualized as a community of practice in CuDA terms, and the social scene of a development meeting in which members of this community gather to conduct communication events of "terumat cod" is analyzed and discussed. Finally, the contextual constraints that shape this event are closely analyzed. This set of analyses confirms the claim that economic rationality may cause specific organizational forms through the application of SFCPs.

Method

In a small essay titled Lessons Learned (2010), ethnographer of communication Tamar Katriel observes that "the fundamental move that governs the logic of discovery in ethnographic work" is one of *encirclement*. She writes:

> Encirclement [is] the product of the kind of attention that turns social scenes into research sites, social practices and events into research topics […] It is often experienced as an intuitive response to things observed and heard, yet often I find in retrospect that it has involved a specific, theoretically-guided kind of noticing, one that creates links between empirical details and forms of abstraction […] It is never just a matter of being in the field — though that is a must — but of being there in a particular way, not in the way of immersion but constantly attuned to its distinctive structural, emotional and aesthetic qualities.

Katriel's description of encirclement as the fundamental move by which ethnographers of communication and CuDA scholars frame their focal objects of observation strongly resonates with Merleau-Ponty's (2004) phenomenological interpretation of Paul Cézanne's painting technique. According to Merleau-Ponty, Cézanne's paintings depict the inherent tension between the conceptual and perceptual dimensions of experience that constitute the relationship between the painter and his object of observation. On the one hand, Cézanne's encirclement outlines the contours of an object in the abstract world of geometry. On the other, it is an open-ended attempt to go around the multiple and shifting features of that object within the spatial and temporal world in which it is presented to the painter.

The works of painting and ethnography are analogous in the sense that both the painter and the ethnographer create links between empirical details and forms of abstraction in delimiting their objects of observation. Much like the geometric forms that direct the gaze of the painter, the communication components that guide the attention of the ethnographer are "abstract entities that exist only in the analyst's descriptive framework" (Duranti 1985:201). In ethnography as in painting, the use of such descriptive units always runs the risk of depriving the objects of their depth, or "the dimension in which the thing is presented not as spread out before us but as an inexhaustible reality full of reserves" (Merleau-Ponty 1964:14-15). The task of both the ethnographer and the painter is therefore to construct a

representation "of an emerging order, of an object in the act of appearing" (Merleau-Ponty 2004:272).

The object whose act of appearing this chapter seeks to depict is the technosocial system of BDFL. The unit of analysis is the event of "terumat cod" (code donation), which functions as the communicative kernel of OSS production within PKW volunteer setting. The descriptive framework is the set of contextual constraints that Hymes (1972a) summarized under the SPEAKING acronym (i.e., Situation, Participants, Ends, Act sequence, Key, Instruments, Norms, and Genres). The usage of this framework in the present chapter below focuses attention only on the contextual constraints of Situation, Instruments, Participants and Act sequence. The reason for this choice is not only that these constraints have greater effect on the shaping of that particular event, but also that they provide clear evidence for the causal relationship between economic rationality and social organization as actualized through the activity system of PKW discourse of work. The normative constraints of "code donation" (i.e., the letter N in Hymes's descriptive framework) are not analyzed here because they require a prior understanding of the group's personhood values. These personhood values are examined in the following chapter, and the normative constraints that they orient are elaborated in chapter 8.

Thesis

The thesis that this chapter seeks to support can be schematized as follows:

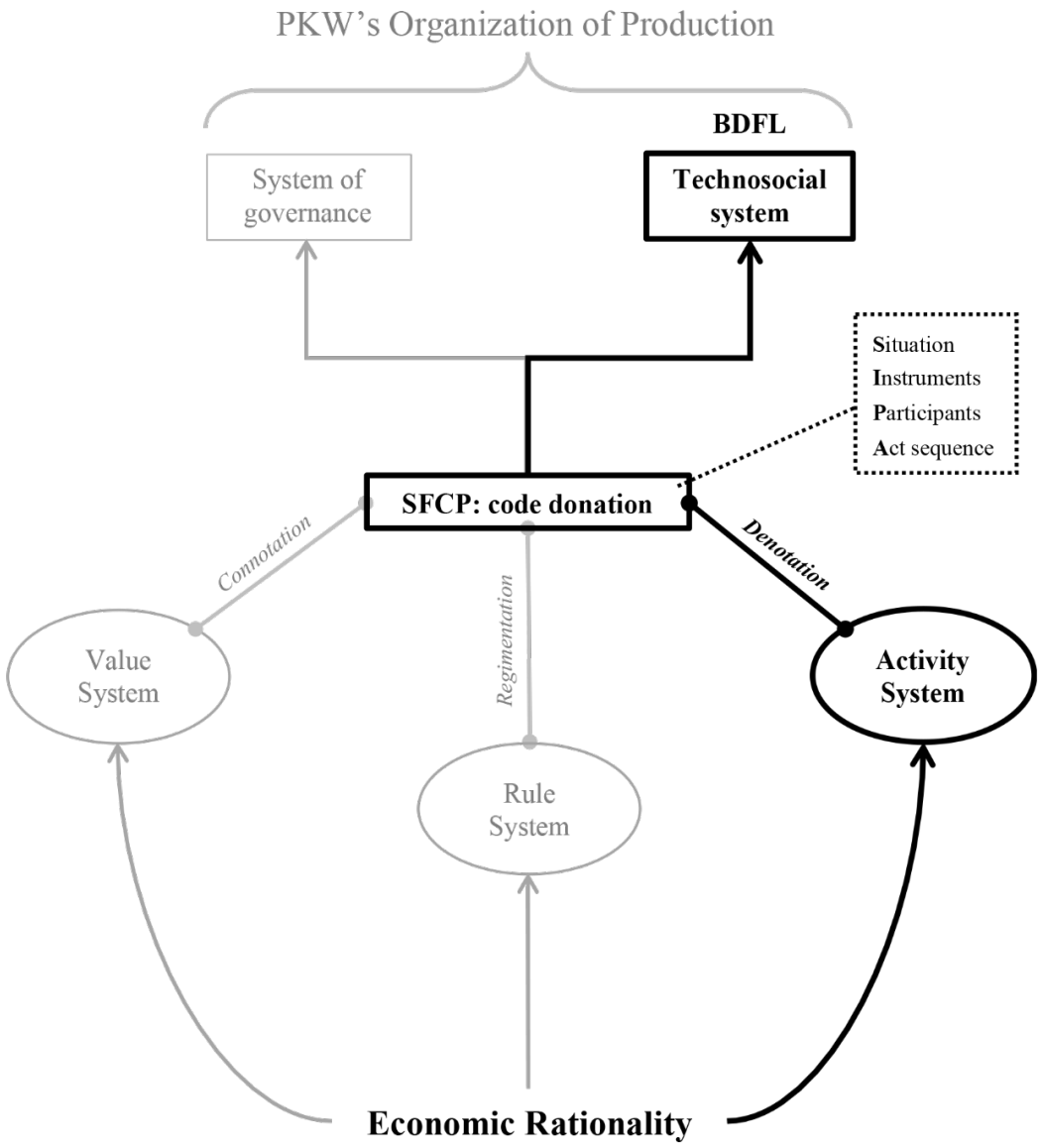


Fig. 10. The Thesis of Chapter 6

In this formulation, the technosocial system of BDFL is constituted by the communication event of OSS production known as "terumat cod." This communication event, in turn, is impressed by the contextual constraints of Situation, Instruments, Participant

and Act sequence. Finally, these contextual constraints articulate and actualize the economic rationality of efficiency optimization (under the conditions of time scarcity and lack of command ability in PKW volunteer setting).

Procedure

Within CuDA, the description of a communication event conventionally begins with some measure of consideration regarding the larger social scene in which it is expected to occur and recur, and the cultural knowledge that individual actors must share in order to coordinate the enactment of such reiterative socially situated events. Within CuDA, the first conceptual entity is denoted by the term "communication situation" (Carbaugh 2007b:3), and the second by the terms "speech community" (Hymes 1972a) and "communicative competence" (Hymes 1972b). To use the evocative language of Clifford Geertz (1983:57), the concept of communication situation is *experience near* insofar as it is considered to be one contextual constraint in the analysis of communication events (i.e., the letter S in Hymes's SPEAKING acronym). By contrast, the deduced concepts of speech community and communicative competence are *experience distant* as they allow cultural analysts to presume the existence of some prior collective consciousness by logical necessity.

The forthcoming cultural discourse analysis thus begins with an experience distant view of PKW as a community of practice. It then describes the situation of a development

meeting where the event of "terumat cod" or some sequential parts of it are expected to recur.[53] It then examines the contextual constraints of Situation, Instruments, Act sequence, and Participants that shape the event of "terumat cod" and the BDFL institution of which it forms a part. This examination concludes with a brief statement on the contribution of this chapter to the accomplishment of the dissertation's overall aim.

PKW as a Community of Practice

As discussed thus far, participation in PKW volunteer setting is comprised of small project teams that organize around the production and publication of civic websites. While these self-managing project teams are entirely independent from each other, their members work for the same official purpose by using

[53] There are two good reasons for treating the development meeting as an independent unit of sociality rather than a contextual constraint upon the event of "terumat cod." First, this communication event does not require participants in the group's eKnights to attend their community's development meetings as it occurs within the confines of GitHub's online platform. The immediate spatiotemporal and social settings of "terumat cod" are therefore virtual rather than actual, which means that the event is situationally conditioned only by the accessibility of participants to computer machines with stable internet connections. The analysis of the virtual constraints that directly shape the event of "terumat cod" is therefore conducted separately, where this event is thoroughly analyzed. The second reason for treating the development meeting as an independent unit of sociality comes from the preference (rather than the necessity) of PKW participants to perform their recurring events of "terumat cod" within its boundaries. By examining this preference, the analysis confirms the microeconomic prediction that volunteers in OSS corporations will have a common interest to optimize the efficiency of their organization of production over and against technosocial obstacles of coordination and scaling, and the CuDA prediction that such an economic interest can only be expressed through publicly accessible media for communication.

the same organization of production. This grouping of eKnights can therefore be considered as a *speech community* or a group sharing rules for the conduct and interpretation of at least one communication practice, and rules for the interpretation of at least one linguistic code (Hymes 1972a:54).

In the case of OSS speech communities such as PKW, it is insufficient for participants to share natural linguistic codes. Rather, participants must also be able to interpret at least one programming code. In the case of PKW, one is required to be able to speak and write in Hebrew and to know how to write in standard American English. Additionally, one must be able to write and read computer code in the agnostic Python language, and to know how to use the Python-based web application framework called Django. If one cannot program in the Django/Python variety, he will not be able to participate in PKW eKnights. Moreover, just knowing how to use this variety is insufficient. One must also be able to write Django/Python code that humans — including the programmer himself — will be able to read and understand over an extended period of time. This requirement presupposes a degree of competency that can only be acquired through months of training in a native programming environment.

Given this emphasis on the ability to write computer code (or "program") and of the instrumental rationale behind it (i.e., the creation of tools), I find it more adequate to speak of PKW as a *community of practice*. While sociolinguists have been using this term in much of the same way that CuDA scholars use the

Hymesian concept of speech community from which it appears to be derived,[54] I use it here to stress the empirical finding that PKW raison d'être is instrumental or substantive across all social levels. As the group's name indicates, "the workshop" is not a place for socialization but rather a semi-industrial arena marked especially for software production. Further, it is clear that participants do not need to get to know each other in person as most of the work is done online and requires only professional and technical knowledge.

To be sure, PKW participants do have a locally recognized discourse of work that had emerged organically as part of the group's official and unofficial processes of institutionalization described in chapter 4. However, this discourse arose for the sole purpose of OSS production in accordance with the principle of efficiency optimization. Understanding the activity, value and rule systems constitutive of PKW discourse of work would thus be unhelpful to potential volunteers attempting to integrate into the group's eKnights if those persons are unable to write Django/Python code at a professional level, an expensive requirement that has been proven to discourage the volunteering of talented programmers with fluency in other agnostic languages such as Ruby and Java.

[54] Widely cited examples are Eckert and McConnell-Ginet (1992) and Bucholtz (1999).

The Situation of a Development Meeting

From a microeconomic perspective, one can expect to find that volunteers in OSS communities of practice work to optimize the efficiency of an organization of production only if the utilities they gain from the participation in that organization of production and/or the use of its software products, exceed their commonly shared disutility of leisure time expenditure that often involves the experience of unpleasant work. The complementary prediction from CuDA is that the social efficacy of any such common economic interest will depend on the usage of local means for communication.

The following analysis confirms these two predictions based on an ethnographic comparison between PKW development meetings and the communication situation that members of the American group Code for Boston call "hack night." The analysis first describes the emergent spatial and temporal boundaries of the two situations based on Merleau-Ponty and Katriel's phenomenological approach, and then uses the conceptual terminology of linguistic anthropologist Judith Irvine (1979) to show that while participants in both groups may be driven by economic motivations, only participants in PKW eKnights have a clear interest to see their projects through to completion, and, hence, to optimize their technosocial system of BDFL vis-a-vis problems of coordination and scaling.

Emergent Situational Focus

In terms of Merleau-Ponty and Katriel's phenomenological language, the period spent between 2011 and 2014 in CFB hack nights demonstrated to me how challenging the move of encirclement can be in ethnographic practice. My focal object of observation in these gatherings did not emerge in any straightforward way from the mass without gaps, in which I was situated as a non-participant observer.

From their inception, CFB hack nights were meant to be calendrical occasions that occurred on a specific day of the week (Tuesday) within a specific time frame (19:00 to 22:30) at a particular place (the Cambridge Innovation Center in the city of Cambridge MA). The general pattern that gives shape to a hack night can be described as an interplay between stasis and mobility. On the one hand, one may observe individuals and small groups of participants sitting around tables that are spread out in the meeting space. These are usually members of local Open Source Software (OSS) project teams who are developing a specific technological tool such as, for example, the Finda website that serves as a way for LGBTQ persons who live in the Greater Boston area to locate service providers and support groups on a map. Most of these participants have laptop computers in front of them. Some are engaged with their screens while others are busy talking. On the other hand, one will also observe a slow but steady traffic of people. Some walk in and out of the meeting space while others walk around the tables or gather together for

varying periods of time in a different location (such as the coffee corner). These participants include the group's organizers, members of the different project teams, and a variety of other visitors and interested parties such as Curt Savoie who served as the principal data scientist for the city of Boston, and Jason, a younger man who identifies himself as a "nerd" with a lifetime interest in cybersecurity.

Against this background, multiple communication units gained salience as they came to my attention. There were smaller situations that resembled an *open state of talk* in which project team members could "initiate a little flurry of talk, then relapse back into silence […] as though adding but another interchange to a chronic conversation in progress" (Goffman 1981:135). There were loosely organized communication events such as casual, professional and introductory talks among the different participants as well as more formal events such as an orientation for newcomers with one of the group's organizers. And there were a variety of programming events such as problem-solving sessions where participants gathered to work on a particular line or section of computer code.

I visited these meetings regularly and at one point also joined a small project team that had developed a mobile application that assists Boston subway users to crowd-source train delays. However, a sense of uncertainty arose in me from the difficulty in locating a focal object of ethnographic

observation. This obstacle encouraged me to extend my exploration to PKW development meetings in Israel.

CFB hack nights and PKW development meetings are not identical but they share a number of characteristics that seemed, initially, to perpetuate the problem of locating a proper object of observation. However, one empirical detail that emerged and caught my attention at the time was the tendency of PKW development meetings to be far more static and rigidly organized than what I had observed at CFB hack nights. In PKW context, I noticed that clear boundaries had been established among members of different project teams and that participants in these groups always worked in fixed locations within the meeting space; this was in contrast with CFB hack nights that appeared to promote participants' movement around the space and interactions among different project teams.

Once I concluded my fieldwork in Israel and conducted a preliminary examination of the data, what became increasingly clear was the significance of the project teams themselves as organizational means within the social situations of a hack night and a development meeting. As I continued to review the perspectives of PKW participants, I found several corroborating testimonies that spoke directly to this ethnographic finding:

(8) Interview (5/1/2017)

I believe that if you will ask the active people what is their frame of reference, so they will tell you. It is the project. And in this sense, the workshop is an umbrella organization for several different and changing initiatives. The projects and the practice areas shift all the time, and this is the main reason for why I don't perceive it as a unified community.

(9) Interview (11/16/2016)

> What you understand very quickly, also in how they talk and also in the volunteering, it's like, there are almost no volunteers in the workshop. Like the volunteers volunteer in a specific project. And from your point of view, when you are in this project, you are not interested in what's going on elsewhere. Not in a bad way, like it's just your identity is an identity of projects. Like you care about the project, when the project goes live […] like on what [the team] works, where [the team] is located [in the production process], things like that.

In excerpt 8, Jonathan, who served in the official role of a community coordinator identifies the social boundaries among PKW eKnights and claims that the primary frames of reference for participants in these groups are their "projects." In excerpt 9, Meirav, who participated in an eKnight for more than a year, provides a clear confirmation of Jonathan's claim.

As I moved to consider my data from CFB fieldsites, I found similar but less explicit views. For example, in one ethnographic observation of the aforementioned project whose members developed a subway train delays application, I loosely documented the following conversation:

(10) Fieldnotes (4/21/2015)

> Sam asks me if I'm doing an MA or a PhD. I answer that it is the latter, and he asks about the kind of perspective that I'm using in the research. I answer that I'm doing ethnography and that my goal is to study the "exotic tribe" of Code for Boston's programmers. Sam says that he took an anthropology class in College, so he knows what I mean. At a later point, Mark comes back into the room and joins the table. I ask Mark if he is in touch with CFB organizers or the members of other project teams. Mark answers that he is not affiliated with the organization at large, and Sam seems to nod in agreement. Mark says he has no idea what the other teams are doing, and Sam updates him about the activities of one such team. Mark tells us that he learned about CFB from another team member, who happened to be his running partner, and says that this is the only social connection that he has, and the only reason for why he is here. Sam says that he decided to join the team when he "shopped for a project" upon his arrival at the hack nights for the first time, and remarks humorously that CFB is made of lots of "small tribes."

As apparent from this excerpt, Mark's account of CFB hack nights combined with Sam's description of "shopping for a project" among "small tribes" fit with Jonathan's assessment of PKW as a segmented community of practice. However, Mark and Sam's comments are less definitive and reflexive. For example, Sam takes care to show that he knows what other teams are doing and his passing remark about group boundaries among different "tribes" is conveyed in a light-hearted and subtle manner.

More generally, I found that CFB participants tended to talk less about, and be less concerned with their organization of production in comparison to their counterparts in PKW. This discursive difference reflected the difference between the spatiotemporal organization of hack nights and development meetings that I attended and observed earlier. While project teams function as centers of gravity in both situations, hack nights were the more fluid of the two. CFB project teams did not attend hack nights on a regular basis, and when they did attend, they did not sit around regular tables. Members of different project teams moved around the space freely and routinely, and the arrival of drop-ins looking to "shop for a project" was much more frequent. It thus became clear that PKW development meetings were more *formal* than CFB hack nights precisely in the way in which "a main focus of attention — a dominant mutual engagement that encompasses all persons present — [was] differentiated from side involvements" (Irvine 1979:779).

If we return to Hymes's (1972a:61-62) original formulation of the SPEAKING acronym, we find that the emergence of a central situational focus (in Irvine's sense) is essentially entwined with the meta-pragmatic activity of ascribing situational and communal Ends to focal acts and activities. Such focal practices with situationally ascribed Ends, in turn, always fall within or between the general categories of social conduct that Goffman (1967:53) called *substance* and *ceremony*. In this formulation, communication practices such as, for example, question-and-answer sequences in a journalistic interview can be best defined as *substantive situational foci* to the extent that the participants who perform them (and their publics) believe that the practices are significant in their own right, i.e., regardless of the manner in which they are conducted. By contrast, communication practices such as the changing of the guard in Buckingham palace can be best defined as *ceremonial situational foci* to the extent that they are felt to have only a secondary significance, having their primary importance as ritual means of communication by which individuals pay homage to socially sanctioned sacred objects (Goffman 1967:54).

This conceptual framework that defines the degree to which situations are organized around substantive and/or ceremonial foci with ascribed situational ends furnished me with means to conduct the analysis presented below.

The focal activities that define the situation of a development meeting are practices of software production. This situational focus is substantive because the practices at issue are believed to be significant in their own right, rather than byproducts of other symbolic actions. This does not mean that the development meetings have no ceremonial features, just that these features are locally and situationally believed to be secondary to the technosocial process of software production.

In Hymes's terms of situational ends, participants in the development meetings share an explicit understanding that the raison d'être of the meetings is to promote the production of civic websites, rather than, for example, socialize with their peers. Indeed, many of the ratified participants in the development meetings told me in almost identical terms that "the people who come here are expected to sit and work." One interviewee, who wrote a B.A. thesis about her participation in PKW, further explains:

(11) Unpublished B.A. Thesis (2016)

> The official purpose for attending the meetings is derived from their name – development meetings. That is, attending in order to write the code and improve it. All those who attend the meetings are clear on what everyone are ought to be doing in them – sitting in front of their computers and writing code.

In contrast with PKW development meetings, CFB hack nights have neither substantive nor ceremonial situational foci because greater emphasis is placed on social engagements in face-to-face interactions. Consequently, and especially in comparison with PKW development meetings, these situations are decentralized, "with

many small groups whose conversations are not meant to concern the gathering as a whole" (Irvine 1979:779). To put it in the words of Harlan, the official "brigade captain" of CFB:

(12) Interview (12/14/2015)

> [What we do here] is not actually coding necessarily. Like sometimes people will code but it will mostly be I'm having trouble with this thing, can you help? rather than, sitting there and actually banging on issues. This is our one night to be together and I think what most of our teams have come to organically is that's the time when we talk about what we're going to do. Not necessarily doing it right now.

Further review of my ethnographic data corresponds with the suggestion that CFB hack nights more closely resemble a "cocktail party" or a "mixer" than a "workshop." The participants in CFB project teams gather not so much to perform task-oriented programming activities as to socialize and network with their peers, public sector visitors, and other interested parties who arrive at the meeting space. In the case of CFB, the relative significance of socializing over coding derives directly from an intent to facilitate collaborations among local government officials and developers who live and operate in the Greater Boston area, and thereby promote the application of technological solutions to public service issues.

This disparity between the situational ends of CFB hack nights and PKW development meetings reflects differing and incompatible conceptions of volunteer participation under the modern work/leisure opposition. For example, hack nights are perceived as easygoing occasions in which one's membership in a project team, and the technical features and civic purposes of that project, provide readymade topics for casual conversation.

By contrast, PKW development meetings are defined as formal occasions for the purpose of software development alone.

The relationship between these oppositional conceptions of voluntary work becomes clear in Harlan's claim that CFB was intentionally designed to accommodate the condition of time scarcity by lowering the requirement for work productivity, and the expectations for the development of "real technologies" and for actual "product delivery:"

(13) Interview (12/14/2015)

I always wanted this to be light. You know? I wanted to create environment where people feel like the work they're doing is good work and productive? But I never wanted to be like- it's not work. So we can't have this like super high intensity sort of like delivery mode? […] Like we are often approached by city partners who want us to work on like- like real infrastructure? Like real technology? And I always tell them no. We can't. I can't commit volunteer labor to that. It won't- you know a volunteer will get sick, or will get busy at work or will go on vacation. And they should. They should. Code for Boston should always be after work, and after home […] This is the third thing on your list. So if I said, you know, hey let's actually work on a permanent system for the city we wouldn't be able to deliver, our people would be sad, our partner would be disappointed, and I would look like an idiot personally.

While PKW founders would have certainly agreed with Harlan that volunteer participation should come "after work and after home," they would, nevertheless, define the temporal frame for that participation in terms of work efficiency so as to maximize the production of "real technologies" and the "delivery" of actual products. Given that a voluntary commitment to work under the substantive standards of one's profession usually costs much more than a situational commitment to the courtesies and ceremonial rules of human interaction, one's preference to work rather than socialize can only be justified on the assumptions

that (i) the utilities of such work are higher than the utilities that one could gain from occasional conversations with strangers; and (ii) the difference between the utilities and disutilities of such work are positive for the developers who chose to engage with it. Based on the principle of utility maximization, microeconomic theory predicts that participants in voluntary corporations will attempt to maximize this positive difference. In practice, such utility maximization can only take the form of efficiency optimization. One way in which this optimization manifests in PKW volunteer setting is clearly heard in the following excerpt where Danny dismisses the idea that the group's development meetings have an intrinsic social value:

(14) Interview (1/12/2016)

1. To treat the workshop as a social activity is to inflate reality. To say that this is really social activity? Pfff! Okay?
2. Also, to come to the workshop in order to look for social life is something that if someone does, I feel a little sorry for him. Because that's not the reason for coming there.
3. […] It's great if people come and hope to get to know interesting people on the way, and end up with friends. I have quite a few people today from the workshop that I treat as friends.
4. But the purpose is not social. In the end, we gather for a reason and on the way we volunteer. So, it should be done in a good atmosphere, and it should be done in a sociable atmosphere. And it happens to occur at the time of the day when we deserve beer. But that's not the purpose.

The participants in PKW eKnights gather for several hours per week to advance the development of their respective civic websites in accordance with their various individual interests. While PKW working space is designed to accommodate leisure time activities and is thus contrasted to the industrial working space of the typical high-tech company, these leisure time activities

are perceived in substantive rather than ceremonial terms. The volunteers who arrive at the meeting space come with the purpose of performing their programming tasks and not to socialize. While they may forge interpersonal bonds of friendship as humans tend to do regardless of the circumstances, most of them engage in social and recreational activities in other leisure arenas with the shared understanding that one does not join an eKnight in order to cultivate his social life or seek the pleasures of companionship. To put this in the words of another core team member: "There are people there who are very sociable […] but when they come to the workshop they just want to sit and program. And that is totally okay."

This situational focus on task-oriented "projects" which may have secondarily "social" consequences as in "friendship" demonstrates and actualizes Bird David's (1997:471-472) original definition of corporate practices as constitutive of "material operations, some of which involve social engagements and relationships." In the typical scenario, PKW participants who spend long working days in their high-tech companies come directly from those industrial locations to renew their working alliances in a similar industrial setting. And while PKW development meetings are marked by leisure time symbols and activities such as the provision of locally brewed beer and the hosting of public lectures on general issues of technological and social interest, they are first and foremost places for software

production where volunteers are expected to “sit and work” at the “regular tables” of their respective “projects.”

The temporal boundaries of the development meetings therefore stand in a peculiar relation with the modern work/leisure opposition in Turner’s (1982) sense. On the one hand, they reinforce the primacy of livelihood making practices over leisure time work in accordance with Maslow’s (1943) hierarchy of needs. On the other hand, they require participants to engage in the same vocational activities that they perform for work at a time of the day when they would otherwise be free to enjoy a period of relief from their jobs.

The same can also be said about PKW hackathons. In order to be effective, such large situations of software production require the committed participation of entire project teams for more than a day. Under the macro-social constraints of the modern work/leisure opposition, these hackathons must (and do) occur only on weekends, when participants have a chance to recuperate from their working week. While accommodating the time constraints of high-tech personnel, PKW weekend hackathons interfere with the participants’ domestic and familial labor obligations. The nature of this interference and the ways in which the group’s administrators have attempted to accommodate it are expressed in the following excerpt, where one community coordinator answers my question about the “average volunteer in the workshop.”

(15) Interview (12/24/2015)

1. Nim: What is the social profile of the average volunteer in the workshop?

2. Sam: ((Giggles)) he is usually a man. He is in his late twenties or early thirties. Surprisingly he has children, most of the time. […] If he has a family, then he usually lives in the suburbs. He usually works in the [high-tech] industry. […] But they are also parents.

3. One of the functions of a community coordinator is to know how to give the community […] better conditions of service. So, one of the things we started to do in the hackathons, which was quite new to Israel […], are baby sitter services. Because in the end, these men- even though they are men, they are also feminists. And they also want to give their women a day off when they are [volunteering]. So, they bring their children, and the kids have a baby sitter.

4. A lot of them by the way will not attend the hackathon because they don't want the woman not to have her day off. Because the hackathon takes place on Saturday, and what, will she stay with the children all the time? Like it was very, very nice to hear parents saying this is our only day with the kids and we will not see them. Which is one of the reasons for doing a hackathon on friday and saturday. It should also be child friendly.

On this account, the typical volunteers in PKW eKnights are family men in their thirties who work in the Israeli high-tech industry and live in the suburbs of Tel Aviv known as Gush Dan. While the development meetings cost them several weekly leisure hours at a time of the day when their young children are usually fast asleep, the hackathons interfere with their parenting responsibilities and the domestic divisions of labor between themselves and their wives. Thus, whenever the administrative staff of PKW organizes a weekend hackathon, it needs (and is expected) to intervene into those domestic labor arrangements so as to prevent the participants' wives from absorbing the costs of their husbands' volunteering.

Although this interviewee's notion of "feminism" is questionable, it calls attention to the fact that leisure time is a particularly scarce resource for any software developer who has

a family and a professional career in the high-tech industry. Insofar as this is true for any participant, it is more so for PKW BDFLs who must give up greater amounts of leisure time in order to lead the production of civic websites within the frameworks of their project teams. A second, related point is the way in which one's participation in a voluntary corporation might affect the non-market economy of one's household. As apparent from this excerpt, the choice to volunteer at a time when one is expected to perform domestic and parenting tasks implicates one's entire familial group, a condition that only increases the core interest of PKW participants and administrators to maximize efficiency so as to minimize voluntary work time.

To the extent that participants in PKW development meetings are motivated to "sit and work" by the utilities they hope to gain from such a voluntary leisure time labor, and that, therefore, the situational ends of these meetings are congruent with the economic interest of utility maximization common to each and every member in the core teams attending them, one should expect to find a negative correlation between the degree to which participants in voluntary corporations are driven by their immediate self-interests and their willingness to engage in a laborious work that could be experienced as boring and unpleasant. Thus, for example, it is apparent that (i) the majority of CFB projects are technically dysfunctional; and (ii) these dysfunctional projects are guided by the external agendas and motivations of public administrators and government

officials. By contrast, it is clear that CFB most successful projects emerged through the motivations and personal initiative of individual participants, who may have performed work during the hack nights, but were not guided or driven by any local production process nor were they influenced by customer demand. In short, these individuals worked and succeeded despite the social orientation of CFB hack nights, not because of it.

Whereas PKW volunteer setting is oriented to satisfy the self-interests of the individuals who attend them, the most hearable collective interest in CFB cultural discourse is compatible with the class interest of participants in the mother organization Code for America (CFA) to promote the employment of software developers in local and national government departments.[55] As is apparent from the data, CFA attempts to achieve this end in at least two ways. First, it runs several different fellowship programs that target excellent graduates of computer science departments across the country. These individuals are selected with an explicit intent to establish the new area of professional expertise known among CFA executives as "civic tech." Secondly, CFA cultivates its local groups of volunteers or "brigades" as potential workforces that could integrate into the field of civic tech once it is institutionalized. To put it in the words of one CFA executive:

[55] This agenda emerged as a corrective measure after the economic crisis of 2008 with the semi-scientific claim that the creation of new administrative technologies will increase the efficiency and reduce the expenses of public services.

(16) Interview (12/24/2015)

To me, […] code for america has two big programs. The fellowship and the brigade. The fellowship, is where you're actually going into cities and building something that the city can actually use and maintain. […] The brigade program, I see as a farm team. […] So, you have people who are learning, they're developing skills, they're understanding the problems (and) how to do civic tech. By building these smaller projects […] they're just tinkering. […] I don't think that governments are going to adopt it, but it's a really great way to hone your skills. […] And eventually, when someone spent enough time at the farm team, whenever there's an opportunity, they get picked up and then become professional civic technologists. We've seen this happen in di ci, where [the founder] of code for di ci is now working for the city of di ci as their chief of innovation.

While PKW participants might create working alliances that advance their individual careers in some way, such alliances are not common, and the creation of job opportunities in the Israeli public or private sectors has no place in the group's mission statement. Somewhat ironically, then, the market oriented approach of CFA is driven, at least in part, by class interests — specifically the interests of the emergent professional class now known in the U.S. as "digerati" — while the principally civic oriented approach of PKW, whose participants typically hold jobs in the Israeli high-tech industry, manifests the utilitarian notion of public interest as an aggregate of private interests. To put this in the words of a co-developer in the Open Knesset eKnight, "I think that in social entrepreneurship it is important not to develop things for them, but for you. First and foremost."

Concluding Remark

The conclusive point here is not that PKW participants are selfish, but rather that they share a common economic interest to bring their OSS initiatives to fruition. If this is true, then

one should expect to find that members of voluntary corporations who come to an agreement that giving up the minimal amount of leisure time necessary for the production of a desired good is not worthwhile for them, will not encounter problems of coordination and scaling, and will therefore have no use of the BDFL institution. Moreover, microeconomic theory predicts that if such producers will find themselves in social situations that provide participants with concrete opportunities to gain more immediate utilities such as, for example, job opportunities or acquaintanceships with likeminded persons, they are likely to leave aside production tasks so as to pursue these other utilities. In such cases, participants are also expected to engage in attempts to optimize or scale the situational means by which those other, more immediate utilities may be gained.

All of these seem to hold true in CFB hack nights where participants are encouraged to socialize without making substantive obligations to collaborate in software projects.

This analysis thus helps to bring the relationship between the explanatory variable of economic rationality and the mediating variable of communication practice to the fore. On the one hand, the descriptive frameworks of Irvine, Hymes and Goffman provide us with essential means to theorize the development meeting as a semi-industrial situation whose local end is the creation of substantive software tools. On the other hand, microeconomic theory provides us with a more straightforward way to explain why PKW participants have designed this particular

situation in this particular way. To be sure, a correct application of the descriptive and interpretive approaches of ethnographers and sociolinguists could have led us, in the last calculation, to the same finding. In this sense, the interpretation of communicative action can be considered as a kind of explanation.

The difficulty here is therefore not the distinction between interpretation and explanation per se, but rather the absence of abstract explanatory principles (such as economic rationality) against which ethnographers can examine their localized interpretations of concrete social action. The need to incorporate such principles into the ethnography of communication was explicitly recognized by Hymes in his original formulation of that research program (1972a:466, emphasis mine):

> If […communication] systems are *generated* or *controlled* […] then discoveries, however fascinating and well described, cannot be explained without a theory that embraces social relations […] If sociolinguistic research often begins as an extension of linguistics, *it must end as an intension of the social sciences* — but in the idiom of disciplines that is only to say that it changes from a way of studying [communication] to a way of studying man as a social being.

While ethnographic studies of communication practices must employ a descriptive-interpretive mode of inquiry to account for the local perspectives of the participants who use them, such studies must not end there. In order to contribute to the understanding of the human condition, localized ethnographic explanations should enter into a productive conversation with the explanatory variables of the social sciences. Moreover, having knowledge of those explanatory variables and how they may relate

to each other and to concrete cases before going to the field may help ethnographers to correctly apply the hermeneutical cycle of interpretation throughout the stages of data collection and analysis — that is, the interplay between concrete explanations of participants and the abstract economic or social logic that lies behind them (cf., Carbaugh and Hastings 1992:158-159).

The Communication Event of Code Donation

My initial difficulty of locating a focal object of observation during PKW development meetings is partly derived in the fact that this communication situation does not provide newcomers with straightforward entryways into the different eKnights. In fact, the only passage that leads to the physical area where project teams "sit and work" at their "regular tables" is blocked by an invisible wall of silence. To cross this socially recognizable borderline is to disrupt the definition of the situation which promotes the activity of software production above all, and thereby to risk embarrassment or a loss of one's public face. The causes of this state of affairs become clear in the following chapters. For now, it is important to keep in mind that the event of "terumat cod" functions as the central communication practice in this situation, that this event is entirely invisible to newcomers because it occurs online, and that participation in this online event within or outside the perimeters of the group's development meetings is the preferred entryway to most eKnights.

The following analysis draws on Hymes's (1972a) descriptive framework to examine the situational, instrumental, participatory, and sequential constraints that shape this communication event. The aim of the description is to show how these constraints actualize the economic rationality of efficiency optimization over and against problems of coordination and scaling in technosocial systems of product development.

The Virtual Settings of GitHub

Today, most OSS projects are hosted and managed on the GitHub platform. This online platform of OSS production builds on and extends the functionality of the distributed revision control system called Git, as originally developed by Linus Torvalds, the creator of the influential Linux project, for the purpose of source code management in a highly distributed environment of software production. The Git application allows participants in an OSS project to store multiple versions of their source code in an online repository. To put it generally, each participant can download from the remote server a copy of the most recent version of the source code to his local machine, work on it independently, and then submit an updated revision to the repository. Each new version that gets accepted into the project is archived separately from all previous versions. This allows team members to (i) easily locate the most recent version for reuse; (ii) review the entire history of modifications that they

have made to the project's source code; and (iii) restore any of the previous versions if necessary.

Originally, Torvalds created Git in the form of a command-line application. Based on this technology, GitHub was developed as an extensive website that provides hosting services for source code repositories in the form of dedicated subdomains (or 'smaller websites'), and a variety of specialized features for project management and social networking.

In the case of PKW community of practice, each BDFL has a dedicated subdomain on GitHub where his source code repository is hosted. This subdomain provides publicly accessible information about the initiative, its activities, and the developers who participate in it.

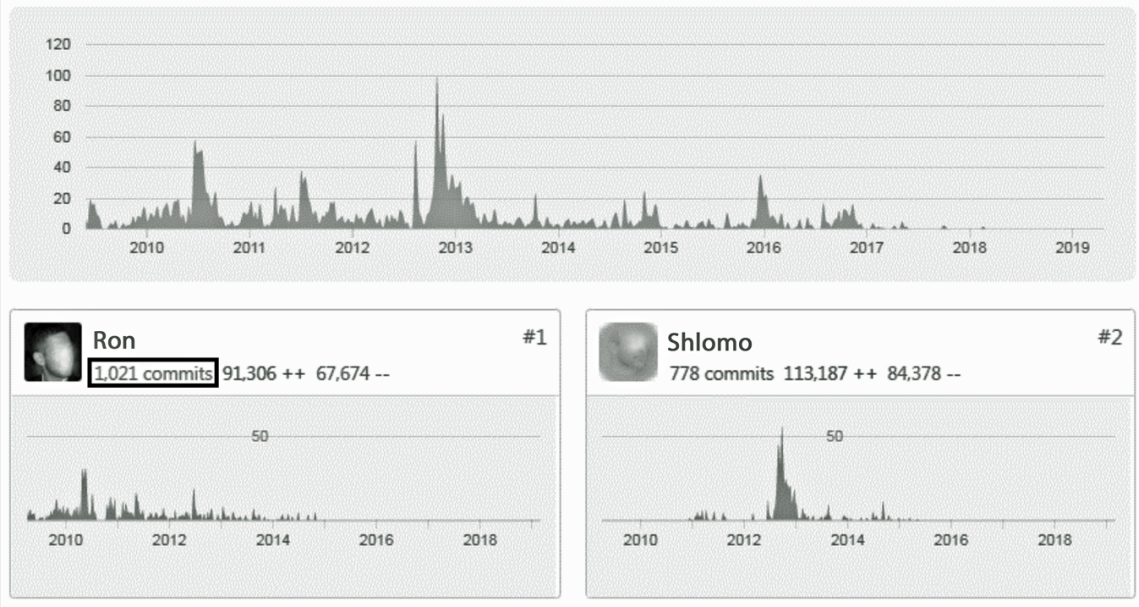


Fig. 11. An eKnight's GitHub Repository

For example, one can easily see the profiles of different participants and how many code contributions each of them

"donated" to the project's repository. Additionally, the subdomain's website provides project team members with analytics about their development process from the point when the first line of code was uploaded to the system.

Fig. 11 above shows how these analytics are represented in the repository of one eKnight. The upper graph in this image provides statistical data about the contributions of that eKnight's participants from the beginning of the project in 2009 until 2019 when this particular piece of data was collected. Beneath this graph, the website constructs a list of all the developers who contributed to the project in the form of a decreasing series. The first person in the series is the developer who made the largest number of contributions (i.e., 1021). Both the name of the contributor and the number of contributions he made are clickable hyperlinks. The first hyperlink leads to the personal profile of the contributor, where others can find more detailed information about his professional background and the projects to which he contributed. The second hyperlink leads to all the lines of computer code that this person wrote and/or modified within the framework of that particular project.

In this way, GitHub accommodates the industrially cultivated habit of high-tech personnel to think about themselves and their creative endeavors in quantitative if not competitive terms of work productivity. While this interpretive framework may not necessarily be taken for granted in any given project, it may

help actualize the attitudes of programmers who already conceive of their vocational personas in such rational-instrumental terms.

However, the real effect of the GitHub platform on the ability of developers with weak social ties to collaborate and work together efficiently over time and across geographic distance becomes fully apparent through a historical comparison with the conditions under which OSS developers worked prior to the Internet era. In Steven Weber's (2004:83) explanation, which is worthy of a lengthy citation:

> Networking has long been an essential part of the open source development process. Before computer-to-computer communications became common, prototypical open source communities grew up in spaces bounded by geography. The main centers in the United States were Bell Labs, the MIT AI Lab, and UC Berkeley. The density of networks really did fall off with something approximating the square of the distance from these geographic points. Extensive sharing across physical distances was difficult, expensive, and slow. It was possible for programmers to carry tapes and hard drives full of code on buses and airplanes, which is exactly what happened, but it was not very efficient. The Internet was the key facilitating innovation. It wiped away networking incompatibilities and the significance of geography, at least for sharing code. As a result, the Internet made it possible to scale the numbers of participants in a project. [Granted,] there are downsides to working together virtually; the transferring of tacit knowledge at a water cooler is a reminder that face-to-face communication carries information that no broadband Internet connection can. But the upside of TCP/IP as a standard protocol for communication was huge because it could scale the utility of electronic bandwidth in a way that physical space could not. Put 25 people in a room and communication slows down, whereas an email list can communicate with 25 people just as quickly and cheaply as it communicates with 10 or 250. As the numbers scale and the network grows, the likelihood of […] pulling into the process people with very different sets of expertise and knowledge — goes up as well.

On this account, the history of computer technology innovation in the U.S. and beyond reflects, and to a large extent actualizes, the rational-instrumental will to maximize the

utility of both the technological means of production and the hardware and software they help produce.

The analytical claim here is that OSS developers who participate in electronically mediated events of "terumat cod" must react to this history of utility maximization whether they like it or, in some cases, are even aware of it. In this regard, the use of durable instruments for technosocial communication is not very different from the use of more ephemeral media such as linguistic dialects, registers, or speech varieties that reflect a cultural history of which speakers are often unaware and in which they nevertheless participate by expressive necessity (Bailey 2007).[56] The same can be said about other OSS means of production that participants in online events of "terumat cod" use, and to which I turn next.

Infrastructural Means of Production

The primary hardware by which participants in PKW eKnights develop their civic websites are their personal laptop computers. To most participants, who spend much of their waking lives in the Israeli high-tech scene, the ownership of at least one laptop as well as a variety of other gadgets such as smartphones and tablet computers is entirely taken for granted.[57]

---

[56] For a lucid account of this complex cultural history, see Thomas Streeter's book The Net Effect (2011).

[57] The expectation that newcomers who arrive at the development meetings will show up with their personal computers is so trivial, that the group's administrators do not explicitly recommend it in the organization's official website. While this may seem obvious, I clearly

Owning a laptop computer is a must, but how do participants set up these machines to create their civic websites? First, eKnight members have to install a source code editor software that provides a framework for writing coded instructions in the agnostic Django/Python variety. The writing of source code necessitates to follow rules of great detail and precision. In the syntax, every letter, numeral, space, comma, semi-colon, bracket and change of line must be used correctly. The smallest mistake in code or syntax creates a "bug" (or a logical error) that may prevent the program from executing (Born 1997:147). Code editors are designed to maximize the efficiency of programming by highlighting the coded instructions in different colors according to the category of terms and syntax rules being used. This highlighting feature allows to easily locate bugs such as syntax errors and improve the readability of the program as a whole. Source code editors may also provide autocomplete features as well as more specialized debugging programs.

---

remember the tension that my non-participant observation brought into the meeting of one project team at an early stage of the research. Equipped only with pen and paper, I sat beside two other participants who were doing something behind one of their computer screens. I directed my gaze at them and wrote furiously in my notebook. In response, they hushed their voices until I could barely hear them. At one point, I felt so uncomfortable that I had to excuse myself to the bathroom. As I looked at these persons from afar, I could clearly see how they regain more comfortable postures. After that episode, I kept bringing a computer to the field and pretended to be busy with my screen while trying to take field notes. While the participants were perfectly aware of this performance, they seemed much more comfortable having me observing them indirectly from a position that appeared to fit with their definition of the situation.

Additionally, eKnight participants must install the Git software that provides them with means to coordinate the development of their source codes. To make things simpler, many participants choose to install a software application called Integrated Development Environment that provides a cohesive unit in which all development activity is done.

While such software applications are necessary for OSS production, the most significant medium for communication and software development in PKW volunteer setting is the Django/Python variety. The utilitarian history that led to the creation of Python and the other agnostic higher-level languages becomes clear in relation to the vertical hierarchy of computer codes discussed in chapter 4. In the following passage, media anthropologist Georgina Born (1997:145-146) uses this logic to make a similar point about the economic rationale behind the more veteran and less portable compiled languages:

> The point about the [compiled] languages is that they provide condensed ways of expressing many thousands of lower-level operations in assembler or machine code. Thus, extremely complicated instructions can be encoded with economy. The rationale is also that they provide more meaningful forms of expression for particular uses. The history of software development, then, has apparently been a search for increasingly technologically and conceptually economical and powerful languages for different kinds of applications.

The more recent development of agnostic languages reflects the same economic principle of efficiency maximization as it is designed to minimize hardware dependencies by means of logical abstraction. Here too, we see how prior historical decisions about the purpose of computer technologies define and limit their

scopes of usage in the present tense. Indeed, it can be said that the very act of software development primes if not coerces programmers to think and work in a more efficient and effective fashion à la Max Weber's iron cage. And if this is true, one may have a good case to examine the relationship between self-love as a human nature factor and the kind of rational-instrumental Zeitgeist that Weber associated with the rise of industrial capitalism in 18th century England.

From the more ethnographic and phenomenological perspective taken here, one can examine the historically determined potential of programming languages for economic efficiency in Carbaugh's (1989:104-108) terms of *communication mode*, i.e., the local ways in which group members perceive one communication practice as more or less direct than other practices, and the relative values that these native participants give to their local ascriptions of directness in speech/action. In this perspective, one can see that participants in PKW eKnights express a clear preference for directness in communicative action that comes into play in their reliance on computer code as a primary medium.

Communication through computer code is perceived to be more efficient than speaking or writing in natural languages for at least two reasons. First, computer code realizes the ideal of communication as information transference in James Carey's (1975) original sense. As such, it allows developers in a software project to minimize their expenditure of work time and emotional energy on secondary, ceremonial or symbolic gestures and other

side involvements. The second reason can be heard in the following commentary by one of PKW BDFLs:

(17) Interview (1/10/2016)

1. I'm very much in the business of show me a code.
2. […] Many times someone can come and say, I think that the map should be red and the blocks should be yellow. Okay ((chuckles)) […] it will be easier if you do some kind of example, a version of it, of a map in red and the blocks in yellow, let me see it and if it looks better then obviously it will get in.
3. […] By the simple fact that you did something good, you contributed to the project in seconds.

To use John Searle's (1976) terminology of speech acts, computer code in this developer's sense entails the substitution of assertions of opinion and expressions of sentiment with declarative acts of creation that follow from the formula: there shall be X, e.g., "a map in red and the blocks in yellow" (L17:2). By writing the literal list of instructions that make up a software feature rather than talking about it, a volunteer may provide a simple and transparent demonstration of his ideas with the potential of "contributing to the project in seconds" (L17:2-3). To "show code" in this way is to realize the potentialities of programming languages for simplicity, literalness, and immediacy. According to media and technology theorist Alexander Galloway, the material and pragmatic nature of these potentialities can be best understood as follows (2004:165-166, emphasis in original):

> [Computer] code draws a line between what is material and what is active, in essence saying that writing (hardware) cannot *do* anything, but must be transformed into code (software) to be effective […] Code is a language, but a very special kind of language. *Code is the only language that is executable* […] The imperative voice (and what philosophers like Austin, Searle, and others talk about in the area of speech act theory) attempts to

> affect change through persuasion but has little real material affect. So code is the first language that actually does what it says — it is a machine for converting meaning into action […] Code has a semantic meaning, but it also has an enactment of meaning. Thus, while natural languages […] have a legible state, code has both a legible state and an executable state. In this way, code is the summation of language plus an executable metalayer that encapsulates that language.

While code may be "the only language that is executable," the idea of a language that "does what it says" has been a fundamental theme of religious discourses in every region of the world. Similar to runes, incantations and the divine language of creation, computer code is not designed to "address an interlocutor" (Bakhtin 1986:95), but rather to manipulate the physical world in which it is enacted. Specifically, programmers write code that manipulates the hardware of their machines to produce specific software artifacts capable of performing certain sets of operations such as analyzing and visualizing the Israeli Parliament's database. These artifacts can then be "shown" to other programmers in both legible and executable states.

The practical possibility of showing one's code to his project team's BDFL rather than talking about it derives not only from the pragmatic nature of computer code but also from the decentralized nature of the Git versioning control system that mediates the process of OSS production. And so, the utilitarian histories of GitHub and Python converge on PKW eKnights to afford rational actors with means to optimize the efficiency of their production processes.

Structure of Participation

While the infrastructural means of software production afford participants in PKW eKnights with potentialities for directness in communicative action, and thus dispose these individuals to productivity and work efficiency, they do not, themselves, overcome the technosocial obstacles of coordination and scaling that face any voluntary corporation under the condition of lack of command ability. Rather, this essential function is reserved to the structure of participation and the sequential organization proper to the BDFL institution. In fact, one can observe that the ownership model of BDFL does not require the use of highly sophisticated technologies for its basic operation. As apparent from Fig. 12, all it takes to become a project owner is to start the creation of some product (call it A) and to have exclusive access to that A.

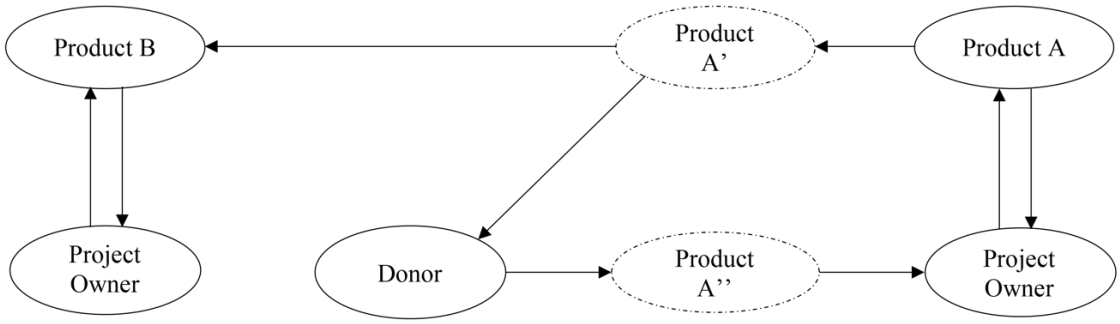


Fig 12. The Technosocial System of BDFL

If these two basic conditions are met, the owner can be sure that others will not interfere with his creative process, and that interested parties who want to contribute to this process will be able to copy, clone or create some representation, blueprint or model of A (call it A') so as to

produce a modified version of it (call it A''), but not to override A with A'' without his explicit permission. While this ownership model prevents others from disrupting or altering one's product, it also allows any interested party to "fork" A' and start a new project from there, a procedure that transforms A' to an essentially different product (call it B) by virtue of project ownership transfer (with the assumption that the new BDFL does make substantial changes to A').

This technosocial system of product development provides members of voluntary corporations with a very simple means to create an authoritative structure of participation under the condition of lack of command ability, and thus to overcome problems of coordination. While project owners cannot explicitly direct their peers, they can examine their peers' contributions relative to the project's vision and to its technical standards, and require contributors to repair their modifications in accordance with these criteria as a condition for their acceptance. Similarly, the BDFL model of ownership allows OSS project founders to scale the utility of a large and dispersed workforce without compromising their standards of quality if they do not wish to do so.

The only limitation of this model's scalability is the project owner's capacity to review and merge code contributions in congruence with the project's growing workforce. In any successful OSS community of practice that manages to cultivate such a growing workforce, there will come a time when the

community's rate of code donations exceeds the BDFL's rate of code reviews under the condition of time scarcity. As history teaches us, the most effective way by which BDFLs overcome this bottleneck effect is the authorization of certain community members to conduct code review procedures. In PKW community of practice, for example, all co-developers have permissions to review and merge code contributions into their BDFLs' products. In other cases, such as Torvalds's Linux project, authorized code reviewers function as a filtering mechanism that allows the project owner to make the final decision with regard to the acceptance/rejection of substantial modifications.

Sequential Organization

The communication event of "terumat cod" in PKW volunteer setting usually involves two consecutive phases that I call *code revision* and *code review*. The preliminary phase of code revision is solitary and relies entirely on the voluntary will of an individual programmer to contribute to the development of a specific eKnight of his interest. Such code revisions are made up of four technical acts and activities. First, the programmer needs to "fork" a snapshot of the most recent version of the original source code from the BDFL's subdomain into a new repository that he sets up within his personal profile's subdomain. From there, the programmer "clones" the GitHub fork into his local desktop and uses the Git software to create a "topic branch" for his personal work on that source code. After

the completion of these two elementary acts, the programmer enters an iterative process of code revision that may be affected by various factors such as the relative complexity of the task, his degree of expertise, the maximum amount of weekly leisure time he is willing to invest in the project, etc.

Within the framework of this revision process, a programmer may fix or slightly modify specific lines of code or write an entirely new list of instructions within his source code editor, but in principle, all the changes and additions that he makes must be "committed" to his local GitHub repository through the Git desktop application. In this sense, the local repositories of project members function exactly like their BDFLs' source code repositories; they allow team members to store multiple versions of the same code and thus to track all the changes they make so as to experiment with different variations of the program.

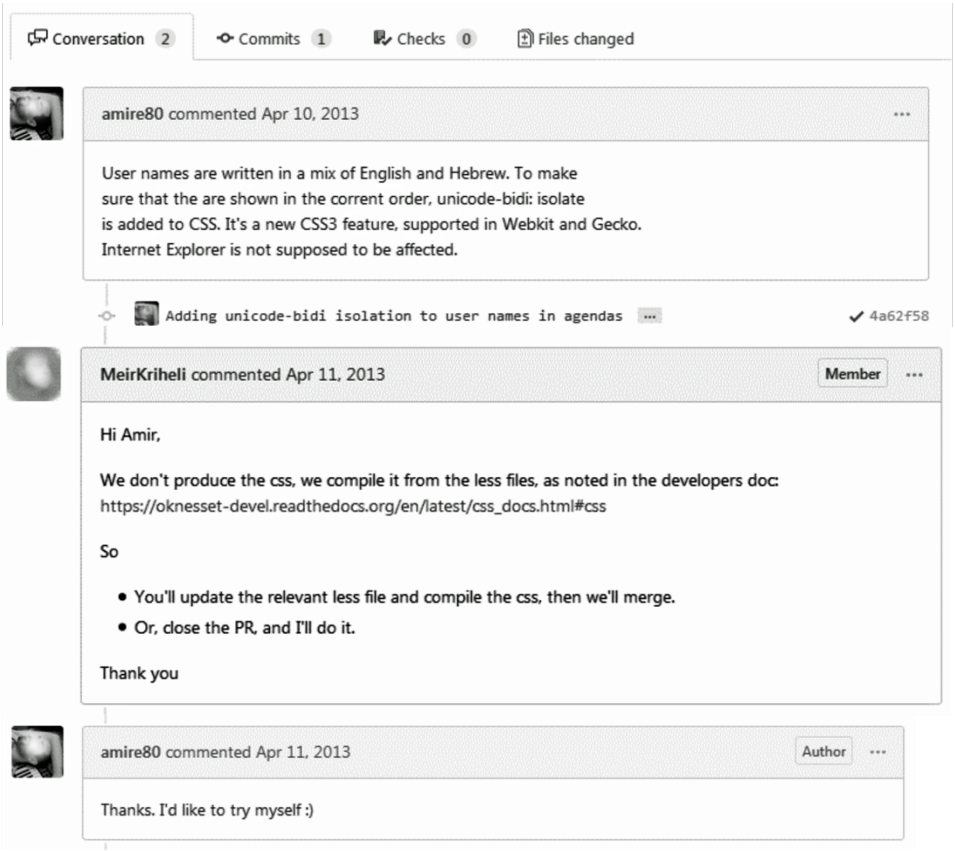
Once the revision work is done, the programmer "pushes" his updated version into his forked repository on GitHub and then commit that snapshot of source code back to the BDFL's repository through a function of "pull request."

This typical procedure may vary based on the project's size and the participants' preferences. Also, some Integrated Development Environments incorporate the Git revision control system and provide user interfaces for managing local repositories as well as better tools for communication among the participants in a highly distributed OSS project. This is significant because individual participants are responsible to

keep in sync with the rest of development since their local repositories are separate from each other. Such software also provide participants with various tools for testing and debugging their source code.[58]

The second phase of code review, which forms the social sequence of "terumat cod," can be observed in the following example from one eKnight project's GitHub repository:

(18) GHOK01



In L18:1, the code donor Amire80 sends a pull request to the eKnight's GitHub repository. First, Amire80 writes a brief description of the problem that his modification fixes and the

[58] Much like in the high-tech industry, the number of "lines of test" that participants in the different project teams write is considered as one important measurement of their products' quality.

technical details of this fix. Second, the GitHub system provides a snapshot of the actual lines of code he revised.

This communication act can be analyzed in two contradictory ways: first, as an act of *gifting* based on the notion of "code *donation*;" and second, as an act of *requesting* based on the notion of "pull *request*." While PKW act of code donation may share some features with the act of gifting as analyzed, for example, by ethnographer of communication Tamar Katriel (1991) or by ethnomethodologist Nick Llewellyn (2011), it is closer to the interactional act of requesting. This interpretation is justified on Marcel Mauss's ([1922]2002) famous claim that any in-person act of gifting is governed by three interrelated obligations: the obligation of S to gift H, and the obligations of H to accept S's gift and to reciprocate it by gifting S (or someone else within the framework of a more complex gift system).

While taking social interaction approaches to the micro analysis of gift exchange, both Katriel and Llewellyn show that local rules of gifting are at least somewhat compatible with the triple norm of the gift in Mauss's formulation. A brief examination of the above excerpt is sufficient to show that the act of pull request is not governed by any such rules. First, Israeli software developers choose to donate code to PKW eKnights not because of some prior sense of obligation, but rather because of the utilities they hope to gain through such voluntary activities of software development, utilities that could but not

at all need to involve philanthropic sentiments or other kinds of prosocial attitudes.[59]

Secondly, the gifted BDFLs and/or their co-developers have absolutely no obligation to accept code donations. It is not only the case that these other core participants choose to create their software projects as OSS projects with the aim of scaling their workforces, but it is also the case that they explicitly attempt to relax the demands imposed upon social actors by the needs of self via the *interaction order* as discussed by Goffman (1967) and Rawls (1987). This intentional attempt to minimize the ceremonial features of human interaction derives from the rational-instrumental imperative of efficiency optimization as discussed in the following chapter.

Finally, while the GitHub platform automatically credits all code donors by adding their profile details to the decreasing series of code contributors (as presented in Fig. 12), the direct beneficiaries of the donation do not offer their benefactors anything in return. Like any conversational request, the act of pull request is directed at a co-participant who can either accept or reject the donation. In L18:2, the eKnight co-developer MeirKriheli thus conditions the acceptance of amire80's "pull request" by the performance of further programming activities on

[59] This analysis can be greatly refined through the usage of Searle's framework for the classification of speech acts. Given that such an examination requires a reconsideration of both the original notion of speech act and the sociological notion of reciprocity, I reserve it for a future publication.

technical grounds. Specifically, MeirKriheli gives Amire80 two options: (i) perform the necessary correction; or (ii) allow him to do the correction instead. In L18:3, amire80 undertakes the revision task and thereby brings the sequence to a close.

The sequential organization of code donation as a communication event can thus be summarized as follows.

Act 1: A code donor *qua* 'benefactor' requests an eKnight BDFL *qua* 'beneficiary' to accept his donation.

Act 2: The BDFL accepts/rejects the request.

Act 3: In the (prevalent) case of rejection on technical grounds, the beneficiary requests the benefactor to deliver a new and better donation.

Acts 2 and 3 may enter into a loop of repetition that could result in an ultimate rejection of the benefactor's donation.

Concluding Remark

As apparent from this analysis, PKW participants attempt – and to a large extent succeed – to override the ceremonial constituents of communication as ritual with the rational-instrumental and substantive notion of communication as information transference (Carey 1975). The success of this attempt reflects a common understanding that voluntary work must replicate the economic rationality of the commercial firm if it is ever to progress into completion and allow individuals to reap the utilities that motivated them to join or create a voluntary corporation to begin with.

Data in support of this finding documents the contextual constraints that shape the event of "terumat cod" as part of the larger situation of a "development meeting." As we have seen, social behavior in PKW development meetings can be best explained by the common interest of participants in the different eKnights to maximize the efficiency of their organization of production. This economic rationale requires a particular mode of expression for its efficacy, which, in this case, is provided by the SFCP "terumat cod." While the communication event of software production that this term denotes can be used in many different ways, any OSS corporation that seeks to optimize the efficiency of its production process vis-a-vis technosocial problems of coordination and scaling will have no other choice but to use it in a similar way to that of PKW eKnights as described here.

Coda

The analysis developed in this chapter provides empirical evidence in support of the study's overall thesis; it shows how the SFCP "terumat code" (code donation) in its capacity as an element in PKW discourse of work's activity system articulates and actualizes the causal relationship between the economic rationality of efficiency optimization and the organization of production component of technosocial system. The following chapter prepares the ground for a complementary analysis in chapter 8 whose aim is to show that the communicative constitution of PKW organization of production's system of

governance is achieved through the regimentation of PKW discourse of work, and that this regimentation, too, follows the principle of efficiency optimization.

CHAPTER 7

VALUE SYSTEM

Introduction

The present chapter has a theoretical aim and an analytical aim. The theoretical aim is to triangulate the finding that social organization tracks economic rationality in the case of PKW volunteer setting, and that this causal relationship depends on communicative action for its articulation. The analytical aim is to enable the examination of PKW discourse of work's rule system in the following chapter.

The triangulation of the central finding of this dissertation repeats the descriptive-interpretive and explanatory moves used in the previous chapter. Specifically, it (i) interprets the communication style known among PKW participants by the SFCP "la'asot cod" (to do/make cod) as an element in the group discourse of work's value system; and (ii) shows that the cultural values that this SFCP connotes can be best explained by the economic rationality of efficiency optimization.

To the extent that the local values connoted by the style of "la'asot cod" help orient the event of "terumat cod" they provide participants with necessary cultural knowledge for the interpretation and justification of the rules that govern the conduct of this communication event. As such, they enable us to complete the examination of the letter N in Hymes's (1972a) SPEAKING acronym in the following chapter.

The central argument of the present chapter is that PKW project owners or Benevolent Dictators For Life (BDFLs) use the SFCP la'asot cod to reproduce a local model of personhood that sanctifies values of assiduousness, proactivity and competence as means to discourage the participation of non-productive volunteers and to reduce the time spent on volunteer reception and integration within the group's development meetings.

The construction of this argument is organized as follows. First, the problem of workforce recruitment that the style of "la'asot cod" arises to solve is developed in detail. This elaboration is worthwhile as it ties together the theoretical threads from the previous chapters. Second, the discussion explains the CuDA framework used in the analytical sections that follow, whose purpose is to show how PKW BDFLs' economic interest of programming time expenditure reduction leads to the creation of culturally sanctioned values and meanings.

The Problem of Workforce Recruitment

PKW eKnights qua voluntary corporations are constrained, fundamentally, by the two conditions of time scarcity and lack of command ability. The condition of time scarcity results from the structural opposition between work and leisure at the foundation of modern industrial society. To volunteer in PKW eKnights is to give up other recreational activities that might serve as mental respite from the type of work that modern actors are required to perform in their professional, domestic and civic capacities day

to day. This not only explains how production time in PKW volunteer setting is scarce, but also why participants in the group's eKnights are constrained by the condition of lack of command ability. PKW volunteers are free from hierarchical structures of command and control, especially from those found in their industrial workplaces, precisely because their practices of OSS production take place at a short duration of autonomy where each of them is required to make decisions freely and perform work of his own choosing.

This requirement from participants in voluntary corporations to exercise their individual autonomy results not from local ideals of individualism, but rather from the material and historical processes that led to the emergence of such ideals in 18th century Europe: an instrumental separation between the realms of work and leisure (Turner 1982), and, conversely, a growing differentiation of the division of labor and the specialization of production among members of disparate groups who suddenly found themselves needing to interact with each other on a daily basis within a variety of urban settings (Durkheim [1893]1984), and consequently, to sustain a public impression of civility in accordance with the principles of equality and liberty (Goffman 1967). To the extent that modern society equates one's free time and one's freedom of choice/action, modern actors have an ultimate right and obligation to do only what they want once they have fulfilled all their other obligations and needs. To put it in Victor Turner's words (1982:36-37):

> Work is now organized by industry so as to be separated from "free time," which includes, in addition to leisure, attendance to such personal needs as eating, sleeping, and caring for one's health and appearance, as well as familial, social, civic, political, and religious obligations […] It is certain that no one is committed to a true leisure activity by material needs or by moral or legal obligations, as is the case with the activities of getting an education, earning a living, or participating in civic or religious ceremonies. Even when there is effort, as in competitive sport, that effort — and the discipline of training — is chosen voluntarily…

Insofar as participation in any voluntary corporation is mostly a leisure time activity, it requires participants to "do what they want" and thereby constitute an ideal type of libertarian relationship where the obligation to respect one's teammates' freedom of action and choice overrides, at least in principle, any consideration of social solidarity. This radical substitution between the capacity of voluntary corporate members to exercise the power of society and their capacity to exercise the power of their own individual wills derives directly from the condition of lack of command ability, that is, the inability of any such individual to take command over his peers by using concrete threats of material or social sanctions. At the same time, it is clear that this substitution between the will of society and the wills of society members is not subject to any individual's personal choice. That is, voluntary corporate members are forced to exercise the power of their individual wills within the modern realm of leisure whether they like it or not. One can therefore speak of a distinct and entirely unstudied kind of social order that arises within the modern realm of leisure to suspend institutional statuses, positions and roles so

as to constitute a libertarian heterotopia where each participant must allow all the others to exist as autonomous individuals par excellence. This moral imperative, which precedes and constrains the self-interested choices of individual rational actors, can be defined in terms of two reciprocal norms: voluntary participation and voluntary selection of tasks. Somewhat ironically, then, the performative constitution of one's autonomous and self-interested individuality within the modern realm of leisure derives, in part, from one's abidance to these two norms and thereby to the authority of society in general.

To the extent that the social contract of voluntary participation and voluntary selection of tasks is a necessary consequence of the industrial revolution, it is expected to be found in any voluntary corporation operative within the modern realm of leisure. That much is apparent from previous research on OSS communities of practice. However, this scholarly literature offers little with respect to the ability of, and the methods by which voluntary corporate BDFLs manage the economic risk of entering into working relations with people who are principally permitted to abandon their tasks and exit the corporations whenever they see fit. Indeed, the necessity for PKW BDFLs to create and use such methods of risk management is apparent from the following two excerpts:

(18) Meir in an introductory talk, cited in an unpublished B.A. thesis (2016)

1. The [BDFLs] are under no obligation to want any code donation that you will want to offer them […] And the other side of the

coin is that you too, as volunteers, are under no obligation to [participate in their] projects.

2. Let's say that I'm the [BDFL] and [two] volunteers sent me solutions to the same problem which are more or less professionally equivalent. [One] volunteer contributes every week and promotes the project, [while the other] troubled himself to send code for the first time. He has been already coming to the meetings for two months. I mostly hear him talking. It's the first time he sent code. At that moment, I want a lot more to preserve [the first volunteer] with me rather than [the second one]. In a sense, I'm even willing to give him up. I might not say this to his face, but there is no doubt that my interest as [BDFL] is to preserve those who contribute more.

(19) Interview (1/5/2016)

We cannot deny you the opportunity to devote your time and efforts to this, but no [BDFL] here will make an obligation to use your code. […It] is very different from the everyday reality that you and I live in. We take it for granted that if we sit and invest effort because someone said that he needs this thing, he will thank us and will want to use it. In the open source world it's not like that.

In L18:1, Meir greets newcomers to PKW development meetings in Tel Aviv with an Israeli "dugri" (or straight) explanation of the contractual agreement between code donors and project owners in this volunteer setting. While project owners cannot force code donors to select and perform programming tasks in what they consider as an efficient and timely fashion, code donors cannot force project owners to accept any of their contributions. Further, as Meir explains in excerpt 19, PKW BDFLs reserve an ultimate right to reject code donations regardless of who the donors are, or, how much time and effort these persons have chosen to invest in the development of their projects' source codes. Clearly, this contractual agreement is designed to protect project owners from the economic risk of leisure time expenditure, while letting them leverage the potential of their open source eKnights for unlimited scaling. Within the role slot

of a BDFL, one can focus on the development of his source code while investing the minimal amount of time necessary for selecting the best among any number of contributions that code donors have submitted to his repository.

The most important consequence of this arrangement is that it forces code donors to share some of the BDFLs risks of leisure time expenditure as they need to consider the possibility of code rejection whenever they think of contributing code to a given eKnight. As apparent from Meir's explanation in L18:2, a primary way to reduce this risk is to display one's dedication to an eKnight by developing that eKnight's source code on a regular and frequent basis. In the case of PKW eKnights, then, project owners favor participants who donate more code in accordance with the basic economic principle of utility maximization.

PKW BDFLs' Problem of Workforce Recruitment

The contractual agreement between code donors and BDFLs described above depends for its possibility on two conditions: first, project owners must be able to attract a relatively large number of code donors over time and across geographic distance; and second, those code donors must be able to integrate into the production process with minimal disturbance of the more substantial work of project owners and their co-developers.

At the time of my fieldwork, neither of these conditions was met in PKW volunteer setting. In contrast with OSS products such as the Python programming language or the Linux operating

system that have a potential to attract programmers who live and operate in different regions of the industrialized world, if only because of their use value (Weber 2004:154),[60] PKW civic tools are limited to the Israeli society and can be best used by local experts such as data journalists, academic researchers and civil rights lawyers. The relatively small use value of such local civic products places a significant constraint upon the group's potential for scaling.

As discussed in chapter 5, it was precisely this constraint that led the founders of PKW to conduct regular development meetings in the high-tech center of Tel Aviv so that their peers and colleagues could come and learn about their projects after having finished work. While this method of volunteer recruitment worked well in PKW foundational years, it became less effective once the group institutionalized and started to attract visitors from all walks of life, people who showed genuine interest in the eKnights but could not donate code and, therefore, had no value to the project teams. And while many other visitors did have the necessary programming skills, they tended to "waste the time" of the group's BDFLs with excessive questions while not offering any "good" or "useful" code in return. In the following excerpt, Elad gives a clear expression of the unfavorable results of such displays of "disrespect" to his volunteer time:

(20) Interview (1/7/2016)

1. Ela: What I'm finding most disrespectful as a volunteer is people who waste my time.

[60] See also ft., 5.

2. Nim: What does it mean to waste your time?

3. Ela: Now this is an important point because it is very painful on a daily basis. […] The actual situation today is that you constantly have to make presentations about the project to people who think it's cool, while knowing that two hours later you'll not hear from eighty percent of them. […] And since I'm both a project leader and the main developer, so all this time I'm in a meeting- and there are periods when the development meeting is the only time I'm available to work on [the project], so like there's no progress and it's very very frustrating.

4. […] When you meet a new person he feels like okay I came to contribute and change the world and I'm like one of one hundred. [And I think,] as if you would stay long enough to try writing a useful code. And that's before I even know if he can write good code.

5. […] The experience of volunteering in the workshop became very frustrating to me once I was dragged into this preoccupation. Once I saw that I invest time in it and it doesn't bring any impact. And then I found myself saying like fuck it I don't accept volunteers. […] And even when I do, I am very unwelcoming. I can say that I started with the approach that one should be welcoming, and I'm still trying to be more communicative. And let's say that my pride is that I do manage to operate volunteers more than other leaders. But it's just wearing out […] and frustrating, and doesn't return itself. […] In this sense, when a new volunteer arrives, it doesn't pay to be nice to him and to invest time in him.

In this interchange, Elad describes the unfavorable scenario of workforce recruitment within PKW development meetings. Notably, this scenario undermines the contractual agreement between BDFLs and code donors that Meir describes in excerpt 18. In the desirable scenario, which can be observed in other OSS projects that run entirely online, a given BDFL learns about the existence of a given code donor only within the framework of code donation. This practical condition functions to ensure that BDFLs will only enter into social interactions with potential volunteers who have already demonstrated their value and ability simultaneously through the single act of "pull request." As apparent from Elad's description (L20:3-4), this

condition cannot be easily reproduced offline when curious programmers arrive at the development meetings in order to see what their peers are working on.

As a narrative protagonist (L20:5), Elad's initial approach was to give such visitors 'the benefit of the doubt' by treating them as skilled programmers who were willing to commit some of their leisure time to his project. Over time, Elad discovered that this approach was uneconomic to the extent that it threatened the existence of his project. The production costs that he had to pay when interacting with people who "wasted his time" were experienced as negative feelings such as frustration and anger. Abiding by the interactional obligation to keep such feelings in check when conversing with people who, after all, "came to contribute and change the world," was itself a form of emotional labor that intensified his frustration. In response to this situation, Elad chose to reduce the risk of leisure time expenditure by giving up the potential workforce of newcomers whom he could not trust.

Guiding Question

It is precisely here where we find the microeconomic problem of workforce recruitment in PKW volunteer setting. On the one hand, the group's BDFLs have an interest in increasing the number of contributions to their source code repositories. This interest is strengthened by the facts that some of the group's more ambitious projects require a minimal number of participants

with certain kinds of expertise, and that the utility of code contributors can be scaled without harming the quality and integrity of the products. On the other hand, any rational BDFL is bound to recognize that the recruitment of programmers in the development meetings is uneconomic due to a widely acknowledged and demonstrated ratio that suggests approximately 80% of visitors — enthusiastic though they may be — will not contribute a single line of code to any of the group's projects. The practical question for PKW BDFLs then becomes, how, if at all, can one maximize his chances of singling out 'real volunteers' among newcomers to the group's development meetings?

Method

The procedure by which the forthcoming analysis answers the question posed above involves a conceptual stage and an empirical stage. The conceptual stage examines the problem of volunteer recruitment from the abstract standpoint of PKW BDFLs qua rational actors so as to predict the most efficient way by which they could solve it. The empirical step tests this prediction against ethnographic data. The conceptual step is taken in this section as it concerns the study's overall thesis. The empirical step is taken in the remaining sections as it involves a close examination of data from PKW volunteer setting.

The Utilitarian Prediction

In order to find the best possible solution to the problem of workforce recruitment in PKW volunteer setting, a given BDFL will first need to observe that the new volunteers who constitute the "twenty percent" of viable participants, and the majority of new participants who enter his project through the online event of "terumat cod" (code donation) belong to the same category of persons. Secondly, the BDFL will have to identify the essential traits that make members of this category useful workers in clear and concise terms. Finally, the BDFL will need to devise a technical task of some kind that will serve as a test to examine the qualities and degrees of these specific traits among newcomers to the group's development meetings.

The personhood traits at issue can be deduced from the economic interest of time expenditure reduction under the limiting condition of lack of command ability. First, any rational BDFL will want to work alongside competent programmers who will not require continual guidance throughout the development process. Hence the trait of *competency*. A second, related, trait is *proactivity* qua the ability to abide by the norms of voluntary participation and voluntary selection of tasks imposed on voluntary corporate members by the modern work/leisure opposition via the condition of lack of command ability. In volunteering alongside such independent and self-reliant developers, a BDFL will increase his own available time to focus on essential programming tasks. Finally, and perhaps most

importantly, is the trait of *assiduousness* or the capacity of developers to take on challenging and tedious programming tasks and carry them out to completion in an efficient and timely fashion. The relative significance of such perseverant industriousness will certainly come to the fore in situations such as the one described by Meir in L18:2. Assuming that the 'ideal volunteer' in an OSS project is a proactive, assiduous, and competent programmer, a PKW BDFL can easily decide if a newcomer is worth his time by directing him to perform an initial 'task-test.' For example, a BDFL could tell a newcomer to enter his project's repository and find something "interesting" or "useful" to do there.

Procedure of Empirical Proof

Within the terms of this dissertation, the expressible personhood traits that make a given newcomer desirable for volunteer participation in PKW can be considered as culturally sanctioned *unquestionables* (Moore and Myerhoff 1977). To the extent that such unquestionables enter the constitution of a local community of practice, they evidence the articulation of the causal relationship between economic rationality and social organization that this study is sought to support.

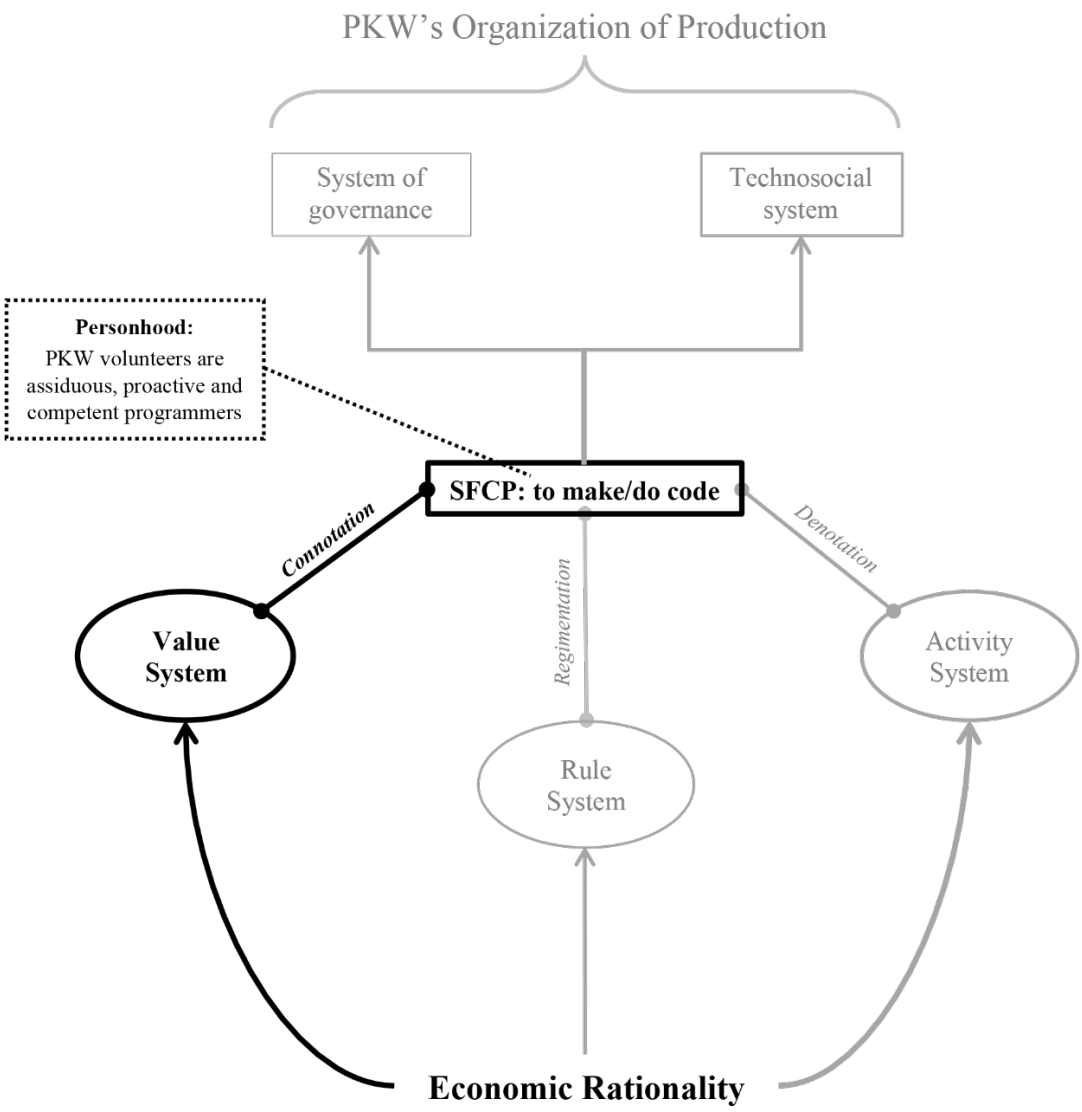


Fig. 13. The Thesis of Chapter 7

The thesis of the forthcoming analysis schematized in Fig. 13 above is, therefore, that the economic interest of programming time expenditure reduction leads the BDFLs of PKW eKnights to sanctify the model of an assiduous, proactive and competent programmer as a cultural unquestionable via the communication style of "la'asot cod" (i.e., to do/make code).

The procedure of empirical proof is conducted as follows. First, the predicted model of an assiduous, proactive and

competent person is examined and verified against ethnographic data. Second, the role of PKW BDFLs in reproducing the expressive force of "la'asot cod" as a mode of personification is examined in some detail. The valued model of personhood that results from this examination enables us to link between the SFCPs of "la'asot cod" and "terumat cod" in their capacities as elements in the activity and rule systems of the group's discourse of work. This analytical synthesis, in turn, is designed to theorize the group's system of governance in the following chapter and thereby to bring the analysis to its logical and empirical conclusions.

To do/make code as a Communication Style

Within CuDA, the interpretation of personhood values must be anchored in the precise Symbol for Communication Practice (SFCP) that connotes these values. In the case of PKW volunteer setting, this SFCP is the meta-pragmatic term "la'asot cod" (to do/make cod). Taken as an element in the activity system of PKW discourse of work, "la'asot cod" denotes a locally preferred communication style which is often contrasted with the pejorative style denoted by the SFCP "la'asot ra'ash" (to do/make noise). This locally constituted opposition between a valued communication style of software production and a devalued communication style of verbal interaction is readily available from the following excerpt by Liat:

(21) Interview (12/24/2015)

1. I learned a lot from Yoav. […] Lots and lots. Because he told me one of the most interesting things.

2. […] He told me, I know that most of the code will fall on some three people. There will be people who come and go, but like […] twenty percent will do and eighty percent will make noise.
3. Like ((chuckling)) there's a huge difference between la'asot cod and la'asot ra'ash.
4. […] And he was right.

In this excerpt, Liat, who did not have prior experience working with OSS communities of practice, indicates that Yoav, who owns one of the group's eKnights, played a significant role introducing her into this kind of volunteer setting (L21:1). On her account, the most significant lesson she learned from Yoav was that the division of volunteer work in the typical OSS project is extremely imbalanced (L21:2). To explain this imbalance, Liat follows Yoav in referring to the well-known "Pareto principle," according to which, one should expect to find that 20% of the participants in a given eKnight will make 80% of that project's source code (if not more) while making 20% of the "noise" (or verbal interaction) surrounding the production process. The other 80% are expected to write only 20% of the source code (if at all) while being responsible for 80% of the "noise" made by the team (if not more).

A basic quantitative examination of the division of labor within Open Knesset, the original project that brought about the formation of PKW in 2011, shows that Liat and Yoav are correct in their estimation. Similar results can be found in every project of such scale; added to which, in most OSS projects, there is

also a positive correlation between the number and the quality of code contributions.[61]

As shown in Table 4 and Fig. 14 below, out of a total of eighty-two developers, six individuals produced 84% of Open Knesset's source code. Sixteen other developers produced 11% of the source code, while the rest of the team, which amounted to sixty individuals, were responsible for the production of only 5% of that code. From here we see how the general economic law that applies to volunteer recruitment within PKW development meetings also applies to the distribution of work among the recruited volunteers. That is, the chance of a BDFL encountering a newcomer at a development meeting who is not only willing and able to volunteer, but is also ready to work as much and as hard as the BDFL does is approximately 4%.

Contribution range	Number of contributors	Number of contributions	Percentage of contribution
500 – 1000	3	2481	66%
100 – 500	3	678	18%
20 – 100	10	352	9%
10 – 20	6	78	2%
1 – 10	60	160	5%
SUM	82	3749	100%

Table. 3. Distribution of code contributions in an eKnight

[61] As political economist Steven Weber (2004:71) has shown, the 80/20 ratio in the division of labor among participants in OSS projects is a factual reality. Indeed, this seems to be the case in every eKnight that manages to attract a large number of contributors.

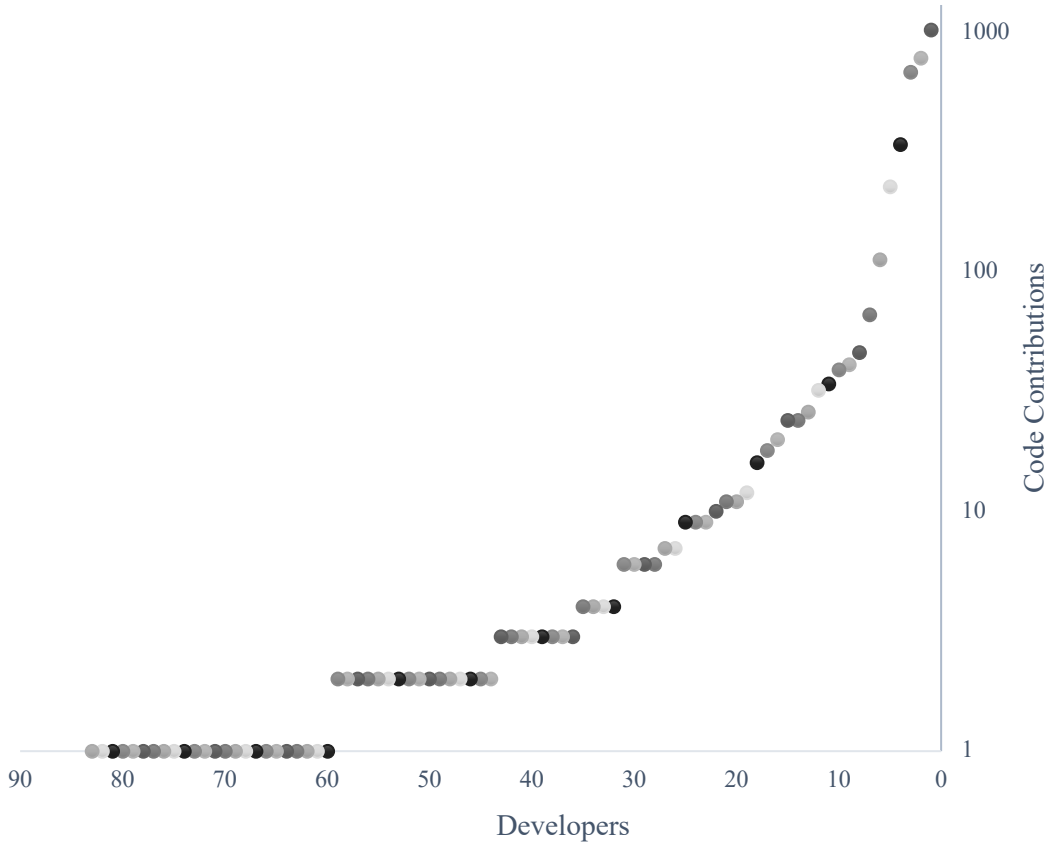


Fig. 14. Distribution of code contributions in an eKnight

This finding directs our attention to the classic microeconomic problem of *free riding* (Olson 1965). Insofar as any rational actor has an inherent interest to consume the fruits of other people's labors, all participants in any given OSS project are expected to sit back and wait for somebody else to work in their place. In this scenario, the technosocial system of product development unravels as no one writes the initial lines of source code that constitute the BDFL institution.

While this problem is certainly valid, there are two reasons for why it does not threaten the existence of most OSS initiatives. The first reason, which has to do specifically with

the motivations of project owners, is that OSS initiatives rarely take form within social settings where the ratified members of a project team are already gathered and ready to begin the production process. To the contrary, the typical OSS initiative emerges as a personal project of an individual developer who seeks to gain some immediate utility. If this utility is gained, no transition to a full-fledged voluntary corporation is needed and so is unlikely to occur. If, however, such an individual developer realizes that he cannot complete the project on his own, or that the costs of such an endeavor are too high for him personally, he will be motivated to publish his source code so as to attract a volunteer workforce to the project.

The second reason that explains why the problem of free riding has little effect on OSS production, which relates more to the motivations of code donors, is that the disutility of sitting and waiting for someone else to perform a desired task is often greater than the disutility of giving up the minimal amount of leisure time necessary for the completion of this one task.

These two reasons bring the microeconomic distinction between the substance and force of individual motivations into sharp relief. While two equally competent developers such as H and S may desire the exact same utility, H's desire could be stronger than S's or vice versa. One can therefore expect to find a positive correlation between the degree to which a given developer desires a specific utility, and the amount of leisure time he will be ready to give up in order to gain it. In other

words, the expectation is that programming tasks within PKW eKnights will not be distributed equally and that this unequal distribution could be partly explained by the relative force of participants' motivations to carry out the projects into completion. This seems to be the case in the Open Knesset eKnight presented above, where the individuals who contributed to the project's source code the most were Elihav, Ilan and several of their committed co-developers.

From here, we learn first that PKW participants are aware of the social and economic conditions of software production under which they move and operate. Secondly, we hear the pragmatic and meta-pragmatic communication practices that mediate the semantic and moral meanings of that awareness. The most productive and dedicated volunteers in PKW eKnights refer to themselves as persons who "do/make code" in contradistinction to their less productive peers who, like the typical participants in CFB hack nights, mostly "do/make noise."

Alternative Labels

Hearable oppositions between interdependent symbols for communication styles, according to Carbaugh (1989:100), are often used as means for the organization and evaluation of other SFCPs that denote locally recognizable acts and events. This is clearly the case in PKW volunteer setting, as apparent, for example, from the following excerpt where Yoav employs the opposition between the styles of "la'asot cod" and "la'asot ra'ash" to differentiate

between proper and improper usages of other communication acts and events within the group's development meetings.

(22) Interview (1/10/2016)

1. There are many people who ask you lots of questions ((chuckles)), they make lots of discussions and you don't see code.
2. I'm very much in the business of show me a code.
3. […] I think that one of the problems of people who come to open source is that they ask for permission […] and feel that they are expected to come up and suggest something, get an approval, go on and do it, and then it will be merged into the project.
4. It's not like that! You are supposed to come and see if there is something here that bothers you, that you think can be bettered. Do it. If you want to propose beforehand it's ok. But […] don't expect to have a prolonged discussion. Do it and send a pull request, and if it is good it will get in.
5. By the simple fact that you did something good, you contributed to the project in seconds.
6. […] In open source, do not ask for permission and do not expect somebody else to do something. Just step forward and hand it in.

As apparent from this excerpt, Yoav defines the SFCP of "showing code" (L21:2) as a proper alternative to pejorative labels of face-to-face communication acts and events such as "asking questions" (L22:1), "making discussions" (L21:1), and "asking for permission" (L22:3). The valued SFCP of "showing code" refers specifically to the first move in the act sequence of code donation as analyzed in chapter 6, i.e., the move through which a code donor requests a project owner to accept his contribution via the Git function of a "pull request" (see also L22:4). In this sense, a volunteer who only "shows code" to his BDFL in the group's development meetings is no different from any code donor who volunteers in an OSS project that runs online.

By encouraging team members to emulate the OSS online mode of participation, PKW BDFLs attempt to secure the few weekly

leisure hours that they allocate especially for the performance of their own programming tasks. As Yoav observes in L21:1, asking a BDFL "lots of questions" about his project, or leading a project team to "prolonged discussions" about their BDFL's source code are undesirable actions because they come at the expense of scarce programming time.

While the economic rationale here is readily apparent, the normative requirement that participants will not ask their BDFLs for "permission" (L22:3) is not as clear. In order to fully understand this requirement, we need to recognize that the act of "asking for permission" violates the contract of voluntary participation and voluntary selection of tasks. This violation has to do with the imposition of interpersonal obligations that not only require an interpersonal sense of mutual trust, but also a degree of institutional protection. If code donor H asks permission of BDFL S to make a specific contribution, and if S grants such a permission, then H is obligated to perform the task at issue and S is obligated to accept H's contribution. However, if H does not perform the task, S has no institutional capacity to punish H or force him to respect his obligation. Conversely, if H submits his agreed upon code to S, and if S eventually decides to reject that donation, then H has no institutional power to dispute that decision.

In contrast with such informal compacts that rely entirely on social solidarity and a mutual sense of trust, the libertarian contract of voluntary participation and voluntary selection of

tasks functions to maintain an equilibrium between the economic risks of code donors and BDFLs. While code donor H can do as he chooses as long as he does not pose an economic risk to the scarce programming time of BDFL S (e.g., by speaking to him), S does not need to attend to the programming tasks that H selects and is, therefore, released from the potential scenario in which he might need to reject H's proposed contribution in a face-to-face interaction (or to "waste time" negotiating with H about the conditions for accepting a contribution that might never be delivered). The contract of voluntary participation and voluntary selection of tasks is, therefore, designed to create a libertarian sense of fairness in a social arena where people cannot be held accountable for their promises and obligations.

Novices who arrive at PKW development meetings tend to "feel that they need to ask for permission" because they have not yet grasped the libertarian essence of the social contract at hand. This frequently leads to misunderstanding due to an unawareness that the only sense of obligation to one's tasks arises "from within the volunteer himself" as some group members put it. We Thus find that the social worlds of OSS communities of practice such as PKW are "very different from the everyday reality that you and I live in" (excerpt 19), not because of a radical transition between cultural perspectives, but because the condition of lack of command ability whose limiting effects on mutual reliance and trust are felt strongly by members of voluntary corporations who operate in the modern realm of

leisure. The economic benefit of the libertarian contract of voluntary participation and voluntary selection of tasks in this arena therefore lies in its ability to eliminate mutual trust as a condition for the possibility of social organization.

Values of Personhood

What is the character of the 'ideal participant' who "makes code" in PKW volunteer setting? The typical answer to this question can be heard in the following excerpt where Yoav compares between valued and devalued models of participation in search of the 4% who only make "good cod:"

(23) Interview (1/10/2016)

1. You are supposed to come in and see if there is something here that bothers you, that you think can be bettered. Do it. […] Do it and send a pull request, and if it is good it will get in.
2. […The best case] is that a person comes and really gets into the code […] And this is the rarest case. And there are people like that. […] The guys I worked with on [a specific eKnight] whom I didn't know beforehand- so let's say that one of them is a superb programmer who is very strong in the field of software testing. And tests are something that programmers don't like to write, and it is hard— there were no tests there. And he came and built for it a proper testing framework at a really serious level. So here is a guy who came in and made a very significant contribution.
3. […] In the best case you have someone who also has the initiative, and he also works and makes the features, and also makes them properly. But this is what everyone is looking for. It's very rare.
4. The worst case is that you come up with ideas. Like you're not doing them. I don't want to say that this is the worst, because many times people came up with ideas, and the ideas were good, and I did them. But […] if you become someone who always comes up with ideas […] and gives suggestions for improvement but do not do anything, so your situation is pretty bad.

By this account, the most valued participants in PKW volunteer setting are "superb programmers" who take the "initiative" to "really get into the (source) codes" of their

project teams with the aims of locating and resolving product issues that "bother them" personally, and of "showing" these solutions to their project BDFLs through the Git function of a "pull request." Accordingly, the least valued participants in PKW volunteer setting are persons who "come up with ideas and suggestions for improvement" with the expectation that somebody else will work in their place.

Competency

The first valued trait that emanates from Yoav's definition of persons who "make code" in PKW volunteer setting is competency, i.e., being a "superb programmer" who can "really get into" an eKnight source code (L23:2), and develop that code not only by "making its features" but also by "making them properly" (L23:3). In PKW discourse of work, such notions of competency are associated with both instrumental and cultural values of self-reliance and maturity. The meanings that link the trait of competency to the instrumental value of self-reliance is heard in the following excerpts where Rony and Yossi try to characterize the kind of person who is likely to "find himself" in the development meetings:

(24) Interview (12/24/2015)

The one who […] knows how to use the stack overflow[62] is the one who will get inside […] The one who needs a template, needs a lot of guidance will not find himself.

[62] In the contemporary field of software development, "stack overflow" is widely recognized as a "question and answer website for professional and enthusiast programmers" whose goal is to "build a library of detailed answers to every question about programming." The website's

(25) Interview (1/10/2016)

You need [to have] technological maturity of coming into the project without anyone helping you getting the code out of [GitHub], to figure out what is going on there, and to be able to put a patch that people could actually accept.

In this discourse of work, a competent programmer is someone who displays "technological maturity" by "making code" without a "template" or a "guidance" from his BDFL (or anyone else for that matter). Such a self-reliant or "technologically mature" programmer helps to maximize the efficiency and quality of his BDFL's production process in various ways, such as the submission of substantial features and code contributions which can be immediately merged into the product (L23:2).

The connotation that links this instrumental value of self-reliance qua "technological maturity" to cultural values of maturity qua "independence" or the capacity to pose and pursue one's ends without relying on other people's help, is clearly heard in the following complementary commentaries by the same interviewees:

(26) (12/24/2015)

People who need constant assistance and attention will not find themselves there. People who are more sole minded and can do it, will find themselves there.

(27) (1/10/2016)

I'm explicitly looking for people who do not need to be held by their hand. [For example,] there was a volunteer who […] complained that he is sitting alone when he goes to the meetings that I stopped attending in Tel Aviv, and he is alone. He doesn't have a table of [our project]. So what am I supposed to do? ((chuckles)) I'm not- like it's

---

content is managed by the users who freely ask, answer and debate practical issues and theoretical problems of computer engineering. According to Liat, participants in PKW eKnights are persons capable of searching for solutions to most of their technical problems within that website's archive, and implement these solutions on their own.

not the boy scouts […] You should already be an adult […] So people who do not need to be held by their hands.

In the first excerpt, Rony draws an opposition between independent or "sole minded" persons who can manage on their own, and "needy" persons who display their technical and social reliance on others by "constantly" asking for "attention" and "assistance." In excerpt 27, Yossi gives a clear pejorative sense to Rony's notion of a "needy person" by describing such a person as an "immature scout boy" who must be "held by his hands" in order to function in the development meetings.

Yossi, who does not live in Tel Aviv where the group's central meetings take place, runs his eKnight online with the expectation that his volunteer workers will be able to manage their individual tasks and contributions entirely on their own. This expectation is premised on the assumption that independence and self-reliance are attributes of "adulthood" whereas reliance on others is an attribute of "boyhood."

To display the character of a "sole minded" person is thus to show others that one has "reached adulthood." In excerpt 27, Yossi relies on this assumption in expressing the degree of independence that he expects from newcomers who seek to join his project. Here, the technological self-sufficiency proper to a "mature" programmer is defined by one's ability to enter an eKnight as a code donor. In other words, Yossi expects that newcomers will not "waste his time" by asking him "lots of questions" or requesting his social or technological "assistance" or "guidance;" rather, he expects them to "show" him acceptable

"patches" or pieces of code that solve specific product issues that "bother them," or that "they think could be bettered" within the online event of a code donation.

Proactivity and Individual Self-interest

While any rational BDFL has a basic interest in recruiting "superb programmers" to his project team, the utility of such experts can only be scaled if they also use "initiative" in finding and solving product issues that "bother them" personally (L23:2-3). Hence the trait of proactivity and the notion of autonomous and self-interested individuality in which it is inculcated. In the following excerpt, Danny gives a clear sense to the relative significance of proactivity over competency in the group's volunteer setting:

(28) Interview (1/12/2016)

1. If you come as a new volunteer, then if you are proactive you will find yourself […] If you are a programmer, it will be easier for you.
2. Now if you are not proactive and not a programmer, you will probably not volunteer.
3. Now, if you are proactive and not a programmer you will find something to do.
4. If you are a programmer and not proactive, maybe someone will find something to do with you. […] When a volunteer comes to the workshop, if he is a programmer, then they try to attach him to a project […] They sit him down to read that project's documentation, or they immediately let him deal with issues of the code. So it's good for a meeting, it's good for two.
5. […] But in order to continue to volunteer in the workshop you must be proactive because over time no one will tell you what to do. You have to look for things to do and do them.
6. […] Insofar as [reactive volunteers] do not find somebody who takes them by the hand […] and gives them some sort of task, so it is very likely that after a while they will simply exclude themselves out.

This commentary invokes the typical situation where a newcomer to a development meeting presents himself as a prospective volunteer. Between lines 28:1 and 28:4, Danny constructs a set of if/then conditionals to argue that the trait of "proactivity" is more important than that of competency. Danny first observes that a prospective volunteer's competency could become handy to the members of a given team with reference to the central role that the social unit of a "project" plays in defining one's place within PKW community of practice.

When a newcomer to the group's development meetings expresses his intention to volunteer, PKW community coordinator "attaches him" to a specific project team whose members then evaluate his ability to contribute to their eKnight. These other participants may "find something to do" with the "programmer" on an ad-hoc basis (L28:4). For example, they might introduce him to their technological infrastructure by "sitting him down" to read the project's documentation, or direct him to solve specific product issues based on their immediate needs. However, the situation in which one is activated by others is unsustainable regardless of one's programming skills. In order for one to volunteer in an eKnight, one has to display his "proactive" character through the style of "la'asot code." The volunteer person must "look for things to do," i.e., define and select his own tasks, and then "do" these tasks.

In PKW discourse of work, the trait of proactivity qua the ability to abide by the rule of voluntary selection of tasks is

conditioned by one's readiness to rely on his individual self-interested will as a primary motivational force. PKW BDFLs assume that their team members know exactly what they want to gain from their volunteer work in order to commit their scarce leisure time and mental energy to software development within the frameworks of their eKnights, and further assume that such individual actors should be allowed to follow their own interests and perform only the tasks they want. As one BDFL puts it, "if [developers] do not get paid, they do what they want; if something does not attract them, then there is no reason for them to do it."

This undisputed belief in the primacy of hedonic motivation as an ontological and moral principle within PKW community of practice is considered by the group's administrators to be a source of organizational trouble, as apparent, for example, from the following commentaries by Liat and Yona:

(29) Interview (12/24/2015)

1. There are issues. Those issues that are [perceived as] interesting they open, and those issues that are not they do not. Like, I will not do what is uninteresting to me. It may be something really really important, but I will not do it simply because it's not interesting to me.
2. […] The approach is that I am a volunteer and therefore I shall do what I want and not what is important.

(30) Interview (1/4/2016)

1. The bulk of what we do are initiatives that come from within the volunteer himself […] Which means that this is something very entrepreneurial that goes very much from the bottom up.
2. And there is beauty in it and there is something wonderful about it. What it lacks is the possibility of someone like myself or even a board of directors […] to come and say okay, we think that it is very important to engage with the issue of health, and then put it into practice.
3. I mean, the whole way of management […] to which I was accustomed, where we looked at the problems, analyzed them and

searched for solutions ((laughs)), decided how we distribute our resources to attack them, is something that doesn't exist here.

4. I mean, [here] it is something very entrepreneurial that really comes from the bottom up […] Here every project is a world in itself, and every volunteer is a world in itself […] But they are not accountable to me ((laughs)) in any way.

In the first excerpt, Liat makes a distinction between two notions of value; the value of a programming task to the volunteer who self-selects it, and the value of that same task to the programmatic goals of PKW (L29:1). She then tries to explain the insistence of PKW participants on the principle of hedonic motivation with the claim that these individuals might think that their choice to volunteer gives them an unconditional right to do as they please (L29:2). However, Liat and perhaps other administrators, such as Yona in excerpt 30, might not perceive the imperative for contributors' unfettered liberty to maintain positive differences between individual utilities and disutilities, an imperative produced precisely because contributors cannot be sanctioned or punished, and because their work will not be compensated in any formal contractual sense.

In other words, PKW administrators might not recognize the norm of voluntary selection of tasks as a condition for the possibility of any voluntary corporation to exist within the modern realm of leisure. Moreover, such administrators might not realize that this norm is the only individually applied means by which the group's BDFLs can protect their interest of programming time expenditure reduction. That is, by letting others do as they want, project owners may also focus on programming tasks of their

choice while simultaneously utilizing the capacity of OSS production for unlimited scaling via events of code donation.

Assiduousness

The last trait that emanates from the SFCP "la'asot cod" is assiduousness, which can be heard in the following excerpts:

(31) Interview (1/7/2016)

1. Our team in hackita zero one- so we were three people […] Yochai, Ziv and I.
2. Yochai was very opinionated and also a good technologist but […] it was very hard for us to work together. Because he aimed towards things that I didn't know how to work with? That is to say, he wanted to develop in angular because it is new and cool […] I wanted to learn django, like to work with django.
3. […] And then it was a bit hard for us. At one point we even worked on the same project on two parallel axes.
4. […] The solution of this was that he simply stopped coming. I mean, he was not determined enough. I mean my determination won.
5. At first it was hard for me because I didn't have a case to be the leader more than Yochai. But once he left, it like solved the problem […] And let's say that Yochai in the points when he dropped back, so because he was no longer involved- so he also accepted my leadership.

(32) Interview (1/10/2016)

You could have a programmer who is not an awesome programmer but he does it. He makes the features. He does not implement them in the best possible way, but here they are. The features are working. So what can you say to him? This is the kind of people you want.

Whereas the majority of PKW eKnights are initiated by one volunteer who then gains status as a BDFL, the participants in hackita program are assigned into ad-hoc projects and are not instructed to constitute the BDFL institution in any of these teams. It is also important to note that these participants are newcomers and are not already aware of the BDFL institution or its role in the OSS organization of production's technosocial system. In most cases one of the participants, usually the most

proactive and competent one, claims ownership over his team's project. In other cases, such as the one described in excerpt 31, two individuals try to claim ownership of a single project at the same time. In Elad's particular case, he and another person who happened to be a more competent programmer entered into a competitive phase in which they had no other choice but to develop the same product on two "parallel axes," each with his own technological preferences and vision for the initiative (L31:1-3). While Yochai was the better programmer, Elad "won" the dual specifically through exemplifying the traits required to bring the task to completion: industriousness, perseverance and self-determination (L31:4). Establishing his leadership role in this way allowed him to integrate the initiative into PKW community of practice as a publicly recognized BDFL.

While this case is unusual, it corroborates Yoav's observation in excerpt 32 that although the ability to write useful code is a limiting condition for participation in PKW eKnights, the quality of that code can be deprioritized to some extent if the volunteer who writes it is an industrious and reliable worker. Moreover, as Jonathan observes in excerpt 33 below, the value of persons who display assiduousness during the meetings of hackita in Jerusalem and/or at the group's development meetings in Tel Aviv, is not only their reliability as volunteer workers but also their capacity for self-improvement à la the modern literary genre known as bildungsroman:

(33) Interview (1/5/2016)

1. […] The will to sit and experiment, to strain and to sweat and to go back to the same problem time after time after time until you succeed to solve it, are skills that one needs to have in order to integrate into communities such as those of the workshop.
2. In contrast with a working place, where you will be fired after the third time that you tried and did not succeed, within the workshop, in an upside down logic, the fact that no one cares if you tried and failed [works to your benefit], because it all happens on your computer, no one even sees it on github.
3. True, there might not be someone who explains to you how to overcome the problem, but if you have the will and the motivation and the mental ability to continue to struggle with this problem, to search in the infinite information and examples available on the web for the means, for the possibility of dealing with this thing, you transform yourself into a source of experience and knowledge and promote yourself.

In this commentary, Jonathan associates the trait of assiduousness with the same instrumental and cultural values of self-reliance and maturity that Rony and Yossi attach to the trait of competency in excerpts 24 to 27. In this locally shared interpretive framework, both "technologically mature" and "technologically immature" developers could be equally proactive persons who rely on their self-interested wills as motivational motors. Jonathan's definitive difference between maturity and immaturity therefore lies not so much in the capacity to write "useful code" as it lies in the capacity to identify and resolve product issues. Whereas "mature" volunteers are free to participate in an OSS production process as they see fit, "immature" volunteers who have not acquired this level of independence or expertise must direct their self-interested wills to vocational maturation in accordance with the principle of assiduousness.

Microfoundations

To better understand the economic rationality and politics pertaining to PKW valued model of personhood, let us first observe that the calculation behind Jonathan's discourse of work in excerpt 33 is that a "technologically immature" volunteer will choose to participate in an eKnight only if the utility he seeks to gain from such strenuous participation is the attainment of a "technologically mature" developer's degree of independence and self-reliance.[63]

From a microeconomic viewpoint, it is clear that PKW BDFLs can only benefit from novices such as Jonathan who come to the development meetings in order to "sweat and strain" toward vocational maturation, as these are the individuals who would constitute exactly the kind of readymade workforce that does not "make noise." However, it is less apparent why the acquisition of technological expertise in this particular volunteer setting is cast as a bildungsroman where the individual achieves independence from his society by practicing and improving his human capacity for autonomous self-learning and growth.

To start answering this question, one must observe, as Marx did in the early 19th century, and as Liat does in excerpt 34

[63] Note that this is the only place in the analysis where a specific individual motivation is considered to be a necessary condition for participation in PKW volunteer setting. That motivation can be theorized as an instance of the will to enhance one's human capital through the acquisition of skills that one cannot easily acquire without monetary costs (Freeman 1997). Interestingly, the central skill in Jonathan's description is not so much the ability to program as it is the ability to acquire and improve one's programming skills in an independent and sedulous fashion.

below, that economic interests — in this case, the interest of PKW BDFLs in time expenditure reduction — can be represented and justified by seemingly unrelated symbols and values.[64]

(34) Interview (12/24/2015)

1. Lia: Those who want to learn python stay, and those who don't simply go away. Speaking of barriers, sometimes the barrier is the python barrier. Because we don't have things- like we have a little java script but we don't have, say, too much ruby. Some people only know how to write in ruby.
2. Nim: Okay, and let's say that a person stays and really wants to go deeper and learn python. So he does it autonomously or-
3. Lia: Autonomously. And if he needs help then usually the guys will tell him to go to stack overflow.
4. And they won't tell him this because they are lazy. They are not lazy.
5. They do want to help him. And their way to help him is to tell him go and read by yourself online.
6. And as a community coordinator, it took me a long time to figure what they want. Like why they are telling him go to read on your own on the internet.
7. But they are right. This is the way to work. I work with programmers today and I know that this is the way to work. Like, before I will ask them why this bug happened, I will check eight times, like on my own, and I will then check on stack overflow, and only then I will go to them and say that there is a bug.
8. Because it saves them time. Like, in the end of the day.

With a measure of caution, one could argue that this excerpt documents the mental process by which Liat attempts to determine why PKW BDFLs do not offer personal assistance to newcomers who struggle with their source codes. She first observes that these experts are "not lazy" (L34:4), and must therefore see some pedagogical value in requiring their

[64] On a methodological level, it is important to emphasize that the interpretive move taken here enriches and deepens the meanings at play in a way that does not violate participants' views of themselves and their practices. In the process, an explanation is developed.

counterpart novices 'to learn on their own how to learn on their own.' Liat then suggests that the experts expect the novices to work autonomously because they want to "help" them; although, she does not say what the nature of that help might be (L34:5-6). Instead, Liat reinforces the value and systematicity of the experts' expectation of autonomous self-learning by appealing to the authority of her own working experience with other software developers (L34:7). Finally, Liat observes that PKW experts want to help themselves as much as they want to help their novice peers, as having one's collaborators teaching themselves how to work on their own secures the time that one allots to product development in both commercial and voluntary settings (L34:8).

And so, we find that PKW BDFLs qua project owners use a culturally desired model of a person who minds his own business with zero reliance on others as a rhetorical means to encourage "technologically immature" volunteers to become, as one participant phrased it, experts who are able to "program like ninjas" in accordance with their in-group 'class interest' of programming time expenditure reduction.

Macrofoundations

While the microeconomic explanation developed above reveals an interesting way to apply the analysis of rational action/choice to the Marxist distinction between superstructure and infrastructure and its key notion of "false consciousness," it cannot, itself, explain why the particular model of personhood

at issue is attractive to these particular individuals at this particular historical moment. In other words, while it is evident that rational action theory can account for the creation of class interest as, for example, Jon Elster (1986) thought, and that such class interest may explain the creation of culturally specific values of personhood as Marxist thinkers have variously shown, it is not as clear why those culturally specific values appeal to the people who aspire to live up to them.

Here we arrive at a necessary juncture between individual self-interest and class stratification in society as this question can only be addressed from a macrosociological approach to the relationship between culture and economy — specifically, the market economy of modern capitalist society.[65]

For example, media historian Thomas Streeter (2011) has attempted to show, much like Max Weber before him, that capitalism as an economic system requires a *spirit* or a cultural unquestionable that mirrors the passions, aspirations and moral and emotional needs of individuals as a condition for its operation. In his words (2001:182):

> Market relations as many economists imagine them are not fully livable. Market and property relations provide at best a crude approximation of human desires for things like freedom, justice, and expression; it is only certain circumstances and confluences of events that allow that approximation to make sense. It is not just a morality tale to say that life needs to be meaningful, that it is not enough to be offered merely monetary rewards. Conceived as a whole way of life, as a complete principled system, then, capitalism is unlivable over the long term;

[65] This line of inquiry parallels Philipsen's (1987; 2002) concern with the inevitable tension between individual impulses and the constraints of macro-social life.

> something more is needed than the calculated drive for profit maximization, which is why people will seek alternatives or seek to articulate the profit drive with other formations. It seems a safe historical generalization to say that, over time, large numbers of people will articulate and seek out forms of life that offer something more or different, forms that are not always nostalgic or backward-looking, forms that can enthusiastically embrace the latest technologies. The exact modes of that articulation can be hugely consequential.

If Streeter is correct in his estimation, it could be the case that the consequentiality of PKW model of personhood, as emanating from the communication style of "la'asot cod," is its ability to moralize the conditions of work in the contemporary high-tech industry. While the present study does not explain the psychological fascination of some PKW participants with this specific model of personhood, it does provide evidence that this model is compatible with the ideal type of worker that contemporary high-tech companies seek in the U.S. and beyond (e.g., Jones et al. 2015:341-342).

This empirical observation should allow us to connect microeconomic theory with traditional Marxism in future research. Moreover, the apparent similarities between PKW valued model of personhood and the ascetic dedication of Protestant Work Ethic that Max Weber attributed to early European capitalism may provide a point of departure to confront between his cultural-historical approach and Marx's historical-materialism (i.e., in examining the directionality of the causal relationship between economic rationality and symbolic representation). Given that neither of these directions help accomplish this study's research aim, I leave them open for the moment.

How do PKW BDFLs qua project owners encourage new volunteers to adopt the group's valued model of personhood in accordance with their own 'class interest' of time expenditure reduction? The first way is simply to raise the bar of entry into the group's eKnights. In the following excerpts, Liat and Ron give a clear expression of this exclusive approach:

(35) Interview (1/11/2016)

It's quite challenging. I mean, most of the programmers in the world are not used to work like that. Even when you are told okay here is a problem go ahead and solve it, you need to learn a lot by yourself. It is very challenging for most technological people because there are lots and lots of things to cover.

(36) Interview (12/24/2015)

1. Lia: Those who want to learn python stay, and those who don't simply go away. Speaking of barriers, sometimes the barrier is the python barrier.
2. Nim: […] Okay, and let's say that a person stays and really wants to go deeper and learn python. So he does it autonomously or-
3. Lia: Autonomously. And if he needs help then usually the guys will tell him to go to stack overflow.

(37) Interview (1/11/2016)

[Any project's] working environment has a manual [and] you are supposed to install it alone. […] There are many people here, many approaches of many people [who say] that those who do not know how to solve all of those issues on their own should not be here.

In excerpt 35, Ron compares the typical high-tech company with PKW eKnights to argue that the degree of independence and self-reliance that the latter require from newcomers is challenging to most professional programmers who hold steady jobs in the high-tech industry. On this account, even in situations where BDFLs such as Yoav instruct a newcomer to patch a specific bug, thereby relaxing the requirement for proactivity, the amount

of independent work and self-learning which is still expected from that person has no parallel in the industry (or anywhere else for that matter). In excerpt 36, Liat corroborates this observation with the example of newcomers who are unable to program in the Django/Python variety, with which the group's project teams compose their source codes.

As explained in chapter 6, Python is the language of web development that defines the technosocial boundaries of PKW community of practice. According to my native informants, a developer who is assigned to solve a code issue in an unfamiliar programming language will have to spend two to three months learning the new language in order to be able to write an "acceptable patch" for that issue. In Liat's observation, the process of programming language acquisition is entirely autonomous as the only sources of information on which newcomers are expected to rely are the "stack overflow" website and its comparatives (e.g., online courses and tutorials).

In excerpt 37, Ron raises the additional problem of development environment installation. In the context of the current discussion, the implication is that even when a newcomer to an eKnight feels comfortable with the Django/Python variety he still needs to overcome the "barrier" that comprises the project's vertical hierarchy of computer code. According to some BDFLs, this barrier can be extremely challenging when the operating system that runs on the newcomer's personal computer is incompatible with the development environment necessary for

running the project's source code on a dedicated server machine. Such obstacles can be bypassed through a range of adjustments and personal expertise that extend beyond that of the average programmer. While the group's BDFLs who know the technical peculiarities of their projects occasionally assist newcomers with such installation difficulties, the general attitude that permeates PKW is that all technical problems can and should be overcome independently. The "approaches" that reflect and justify this attitude, to which Ron refers in this excerpt, speak to the rational-instrumental principle of efficiency and the cultural unquestionable of self-sufficiency.

Character Tests

The second way by which PKW BDFLs encourage new volunteers to work independently in accordance with their economic interest of time expenditure reduction is the conduct of simple tests of character. In the following commentaries, Yoav and Elad give a sense of how such examinations take place in their respective project teams:

(38) Interview (1/10/2016)

1. Nim: In terms of getting new volunteers to the project, you have- again I keep doing this contrast with the industry. So there you have a job interview. Here you don't exactly have it, and people come to you, you need to filter them, how-
2. Yoa: […] It's very simple. You give someone a task and he either does it or doesn't.
3. […] I had this case when someone came to volunteer [in our project] and he said [to me] I'm a programmer and I know how to do all sorts of things. And this- and somehow something about him- I couldn't get if he really was a programmer. I had no idea. So I said okay, here you go.

Here's a task. Do it and we'll go from there. And he didn't do it! So that […] was all that was needed.

4. […] In a job interview, let's say of programmers in [my company], we give them a task that they need to write. And according to this we examine them. You could say that here it's something similar. [But] the task is something real that we really need. Not something we invented. And if the guy went and did this task, great. And it's not trivial. Like the people who really do it are rare.

(39) Interview (1/7/2016)

1. I can tell you what my process is. Let's start from point zero. If the [community coordinator] asks me if I need a programmer, I'll try […] to hint to her or tell her to find out for me if he knows something. […] And then sometimes I say no at this point.
2. But assuming that I do meet a volunteer [when] I'm there, […] then I make a brief presentation of the project and [describe] the directions a little bit. I try to […] feel the person and understand […] how much he brings with him. [I] also try [to do] an initial coordination of expectations. Like what is expected of him […] And I make it clear to him that he has to work alone.
3. I have a kind of regular mail that I send. […] Mainly a list of links to the project. Like this is the code, this is that. In the next step I say okay you should know django. Do you know django? Usually the answer is no. I then send the tutorial. The tutorial is a lovely trap because this is usually the time when [I] don't hear from them anymore and that's where it ends.
4. […] Now a volunteer whom I feel will be worth the investment, so I also accompany him more […] I will also ask how he progresses and I will help him solve problems.
5. Volunteers whom I less believe in, so on the contrary, either I will ignore them or I will be laconic in my conduct. Like in my responses. And there were volunteers I rejected like that.

As apparent from these excerpts, both Elad and Yoav tend to relax the requirement for proactive and self-interested action when they meet a newcomer at a development meeting for the first time by requesting that person to perform some task. In Yoav's case, the task had to do with a real need of the project team. In Elad's case, the emphasis was more about the acquisition of programming skills necessary for the participation in the project; although, a volunteer's command of similar knowledge is also tested in Yoav's case.

The point of these tests is to ensure that the newcomer is "worth the investment" of the BDFL's scarce programing time (L39:4). The central personhood trait whose veracity is examined in this way is assiduousness. Economically, the relative significance of this trait is readily understandable as individuals who perceive themselves as undergoing a process of vocational maturation constitute a reliable workforce whose members are busy scaling their programming skills. By contrast, volunteers who do not display assiduousness will not receive welcoming attitudes from the group's BDFLs and the other core team members (L39:5).

We hereby arrive at a clear confirmation of the prediction from microeconomic theory that the ideal volunteer in PKW eKnights must be an assiduous, proactive and competent Django/Python web developer. Accordingly, we find that culture and communication can be used as means to moralize and dignify rational calculations of utility maximization on both micro and macro sociological levels.

CODA

The analysis presented above sufficiently accomplishes the theoretical and analytical aims introduced at the beginning of this chapter. Theoretically, the analysis confirms the thesis that social organization tracks economic rationality in the case of PKW volunteer setting, and that this causal relationship

depends on Symbols For Communication Practice (SFCPs) for its articulation.

More specifically, the analysis shows that the personhood virtues associated with the SFCP "la'asot cod" (to do/make cod) in its capacity as an element in PKW discourse of work's value system can be best explained by the group project founders' economic interest in leisure time expenditure reduction. While the analysis does not detail how, exactly, these local values of personhood enter the communicative constitution of PKW organization of production's technosocial system and system of governance, it does provide evidence for the existence of such a process of structuration (e.g., by explicating the rite of passage that newcomers must complete in order to integrate into the group's community of practice).

The following chapter, which focuses attention on the organization of production component I call system of governance, is aimed to provide such detailed evidence. Hence the analytical significance of the present chapter to the overall finding of this dissertation. To the extent that the event of "terumat cod" (code donation) is the communicative kernel of the BDFL institution, the personhood virtues associated with the style of la'asot cod must serve in the justification and moralization of the rule templates or protocols that give it shape and credence. In providing an in-depth examination of these rules relative to the personhood virtues elaborated here, the following chapter completes the Hymesian analysis presented in chapter 6. In so

doing, it provides a view of the precise ways in which the economic rationality of PKW participants determines the organization of production components of technosocial system and system of governance via the local SFCPs “terumat cod” and “la’asot cod” and the discourse of work of which they are parts.

CHAPTER 8

RULE SYSTEM

Introduction

Chapters 4, 5 and 6 of this dissertation have shown that participants in Open Source Software (OSS) voluntary corporations such as PKW eKnights use a readymade political institution called Benevolent Dictatorship For Life (BDFL), and that BDFL is the only political institution capable of overcoming the technosocial problems of coordination and scaling proper to the OSS organization of production. Following this analysis, chapter 7 has demonstrated how PKW BDFLs managed to create a culturally valued model of personhood around their shared economic interest of leisure programming time expenditure reduction, and how the application of this model provides a practical solution to the problem of volunteer workforce recruitment.

The present chapter converges these two lines of analysis on the OSS problem of project governance that speaks to the conceptual and practical tension between the regime-types of democracy and autocracy in their purest forms. The specific aim of this consideration is to show how PKW BDFLs use the culturally valued model of personhood that reflects their economic interest of time expenditure reduction as means to justify and legitimize a meritocratic system of governance that they call dookratia, and in which one's decision making power over product development is

measured by the overall percentage of source code that he has contributed to the creation of that particular product.

With this finding in hand, the analysis brings this dissertation to its empirical conclusion. Until shown otherwise, any rational actor who starts an OSS corporation under the conditions of time scarcity and lack of command ability will first need to create and implement an institution that functions in some similar manner as the BDFL. Secondly, any BDFL's economic interest in time expenditure reduction will lead to the creation of a culturally valued model of personhood that sanctifies traits such as those found in PKW volunteer setting. Finally, any multiplicity of individuals who show role alignment with this model of personhood will form a polity that largely substitutes rhetorical acts of persuasion with programming acts of creation.

The analysis developed below is separated into three parts. The first part constructs the OSS problem of project governance relative to the oppositional regime-types of autocracy and democracy. The second part accounts for the regime-type of dookratia by which PKW participants mitigate the tension between these two poles. The third and final part examines the Symbols For Communication Practice (SFCPs) "la'asot cod" (to do/make code) and "terumat cod" (code donation) as elements in PKW discourse of work's rule system so as to demonstrate how these practices regiment the group's values of personhood and the meritocratic system of governance to which they give rise.

The Problem of Project Governance

The problem of project governance in PKW volunteer setting concerns the incapacity of the group's BDFLs to make obligatory decisions and to resolve conflicts among their peers. The decision making and conflict resolution referred here emerge from practical questions that have no clear rational-instrumental answers, such as, for example, questions about mutually exclusive directions of product design. In situations such as these, voluntary corporate's members are bound to experience the irresolvable tension between the regime-types of autocracy and democracy, a conceptual tension that sets the terms for any human system of governance. The following discussion thus poses the general problem of project governance through an examination of the failing attributes of both autocracy and democracy and the untenability of these regime-types as systems of governance in any voluntary corporation.

The Limitations of Autocracy

As explained in chapters 4 and 6, an OSS project's BDFL can enforce an organizational order upon participants in his production process only by means of gatekeeping. That is, a BDFL can control which code donations will enter his product, and under what conditions or terms. This dependence on the negative force of gatekeeping means that no BDFL can dictate his vision for product development in a unilateral way, and especially so in situations where the other team members do not understand his

rationale or disagree with his decisions. Conversely, no BDFL can give or determine rewards, as participants in OSS production assess the value of their own leisure time expenditure.

That said, it is certain that a unilateral dictatorship is both a conceptual and practical impossibility regardless of the condition of lack of command ability. As anyone who has ever tried to play the role of a dictator knows, people — old and young — can be commanded and controlled by force only to a certain degree. Once this degree is reached, members of the homo sapience (as much as members of other animal species) will start to express dissent, which, in turn, will require a dictator to apply more coercive force, thereby consuming additional resources and time. Anyone who tries to run a production process in such a manner is therefore bound to impede the overall efficiency of that process. Between this essential political-economic problem that inevitably presents itself to any autocratic system of governance and the condition of lack of command ability that defines the modern realm of leisure, it is clear that no rational BDFL will choose to rule by coercion alone even if he had the power to do so.

The Limitations of Democracy

The fact that voluntary corporations cannot be ruled by autocratic coercion does not mean that they can be ruled by democratic consensus. Insofar as project team members self-organize in the modern realm of leisure with the aim of creating

a functioning piece of software and so need to prioritize their programming activities over any other kind of social engagement, they are bound to find, at a very early stage, that their economic interest of time expenditure reduction contradicts the basic temporal requirements of consensus making in deliberation.

While this contradiction derives directly from the condition of time scarcity, it is also the case that a true democratic regime, like its autocratic counterpart, is untenable. A total commitment to the principle of rule by consensus will force a BDFL to respond to any event that requires some practical decision by transforming his project team into an ecclesia. Under such a radical requirement for consensus making in deliberation, any material process of production will inevitably come to a halt. Thus, running a production process only by means of democratic deliberation and consensus making is not an option.

The Limits of Constitutional Autocracy

Finally, one must consider the viability of a constitutional autocracy where a BDFL is expected to declare a decision unilaterally in some situations, and to defer a decision to the other team members in others. Irrespective of one's vantage point, the collectivity in OSS communities of practice must embrace the fact that any BDFL can assume total authority over project related conflicts and disagreement by virtue of his ownership prowess alone.

At the same time, participants in voluntary corporations who live and operate in liberal democratic societies are likely to expect a certain degree of influence over product development, an expectation that requires the application of some democratic process of consensus making as a condition for its fulfillment.[66] Indeed, it is apparent that PKW eKnights do involve both democratic and autocratic forms of governance. This, however, does not solve the problem of project governance as it is equally apparent that PKW BDFLs rarely use any such autocratic or democratic forms of governance. This should not be surprising as the foregoing discussion has shown that both autocracy and democracy are detrimental to any organization of production.

To the extent that the space between autocratic and democratic forms of decision making under the conditions of time scarcity and lack of command ability creates an essential problem of governance for any voluntary corporation, and given that many such corporations exist and prosper regardless of this problem, OSS production must be governed by an entirely different regime-type, one that can, by its very structure, reduce as much "noise" as possible while simultaneously producing as much functioning source code as possible in accordance with the principle of

[66] This expectation is also likely to intensify under the normative requirements for voluntary participation and voluntary selection of tasks enforced upon volunteers in OSS production by the condition of lack of command ability and the work/leisure opposition from which it is derived.

efficiency optimization. This system, which PKW participants came to call "dookratia," is examined in the following section.

Dookratia as a System of Governance

It was precisely the discovery that a democratic rule by consensus is untenable which led Elihav, the BDFL of the original Open Knesset project that heralded the creation of PKW, and his co-developer Ilan, to apply the regime-type known among OSS developers, and especially among members of the Debian project (i.e., a specific "fork" of the original Linux project), as "do-ocracy" (Halchenko and Hanke 2012).[67] In Elihav's words:

(40) Interview (1/30/2017)

1. Nim: So how do you govern a project […] that becomes more and more complex and manage the work or create some form of organization?
2. Eli: So first of all we tried to make it very democratic because it sounded appropriate to the nature and purpose of the project.
3. But we realized very quickly that it doesn't work. Like, things need to be done.
4. And like people- once there is a line, people understand or see the agenda from the operations that happen. So it is very easy for them to decide whether they are interested to join or not to join.

[67] It is important to note that the model of dookratia as used by members of OSS communities of practice and other self-organizing groups of volunteers is often described in libertarian terms of voluntary association. This much is evident from the entry "DoOcracy" in the Community Wiki website at https://communitywiki.org/wiki/Do-okratia:

> A do-ocracy […] is an organizational structure in which individuals choose roles and tasks for themselves and execute them. Responsibilities attach to people who do the work, rather than elected or selected officials. The term is popular with libertarian management aficionados. […] Doing a task is in itself justification for you being the person who does that job.

5. I remember that we even had discussions about it. At some point we invented a name for it, what we called dookratia from the word la'asot [to do/make]

6. […] Like it is open source, and again I didn't know- Ilan had more background in the open source community, I had much less. I learned it while working. But I became very attached to it. Like really.

7. Nim: So you say that this idea of dookratia actually comes from the tradition of open source?

8. Eli: Yes, totally.

In answering my question about the organizational features of Open Knesset (L40:1), Elihav explains that his and Ilan's original intent (as referenced by the pronoun "we" in L40:2) was to establish the project on democratic principles. The rationale for this decision had to do with the fact that unlike the majority of OSS projects whose non-proprietary software have only little implication for civil society, the Open Knesset website was created with an explicit intent to affect the way in which Israeli citizens engage with their parliamentary representatives. Elihav and Ilan's attempt to "make the project democratic" was therefore premised on the idea that the project's goal should be congruent with its form.

However, this attempt was largely unsuccessful due to the condition of time scarcity (L40:3). On the one hand, the project became more ambitious and complex with the addition of competent web developers. But on the other, the increase in the number of participants added more voices that needed to be taken into account in every step of the way. It was therefore not long before Elihav and Ilan realized that their team spends more time on discussions than on software production, and that while

everyone has something important to say about their intended contributions, pending programming tasks remain untouched.

As Elihav implies in L40:4, the initial idea to replace the democratic model of Open Knesset with some other regime-type came from the observation that declarative acts of programming create durable material realities that cannot be easily undone, and that these realities place limits on the directions that further product development may take.[68] In the case of Open Knesset, the realities at issue had not only to do with the technical features of the project's source code but also with decisions about the construction of the Knesset database and the analytical functions that the website operates on this database, the OSS licensing of the project, and the decision to not finance the project through advertisements.

When a new participant came to the development meetings, he was therefore able (and expected) to "see" the BDFL's agenda from the project's source code, its legal and non-commercial features, and the specific ways in which the website allowed Internet users to interact with the Knesset database. This, in turn, enabled him to quickly form an opinion about the project, and decide if and how he wants to contribute to it.

[68] This observation provides a concrete evidence for Langdon Winner's (1980) famous argument that material objects may 'have politics' in the sense that their creators can (but do not must) use them as durable symbolic means to fix specific social and power relations, which, in turn, may enter as constraints upon their situational uses by society members in future events.

By allowing their agenda for product development to 'speak for itself' in such a way, Elihav and Ilan confirmed that technological and civic ideas can be effectively encoded into a software artifact rather than talked about, and that any form of communication beyond that artifact (in its legible and executable states) are redundant and, therefore, inefficient.

In L40:5, Elihav describes how this realization led him and Ilan to form an alternative model of governance for Open Knesset based on the OSS notion of "dookratia" (do-ocracy). Under the condition of time scarcity, the advantage of dookratia (as means for efficiency optimization) in mitigating the tension between the extremes of autocracy and democracy is that participants can take initiative in contributing to the project however they see fit rather than "waste time" or "make noise" in discussing their ideas or points of view.

Once a specific "code donation" is proffered, and only if this contribution is controversial, a BDFL qua "code reviewer" can quickly decide whether to merge it into the project's source code based on the democratic principle of rule by majority, or, in less frequent occasions, based on the OSS autocratic principle of rule by ownership right. Given that any iteration of the software artifact establishes firm realities and thus places limits on the production process, industrious team members who offer a large number of code contributions on a regular and frequent basis have a relative advantage in actualizing their personal visions for the product and in influencing the

trajectory of its development. In the following excerpt, Yossi neatly summarizes this essence of dookratia as a decisionmaking procedure, or, more precisely, as a procedure for decision making prevention and noise reduction:

(41) Interview (1/10/2016)

It is very much oriented to the side of doing. The one who will do, the things he wants will happen. And the one who will come up with ideas […] even if he is terribly smart, it will almost never happen. It is like that in any such project.

Dookratia as a Regime-Type

PKW BDFLs' decision to adapt the OSS regime-type of dookratia replicates the Marxian logic by which these individuals converted their economic interest of leisure/programming time expenditure reduction into a set of locally valued personhood traits. In both cases, we see how the same minority of project owners expresses and justifies its common 'class interest' through the articulation of local values and meanings. Given that the regime-type of "dookratia" and the personhood model of an assiduous, proactive and competent Python/Django web developer reflect the exact same class interest, it should come as little surprise that the two cultural unquestionables complement and require each other. If we follow the logic of PKW valued model of personhood to its political conclusion, then, we find that status and power within the group can be best acquired by means of one's demonstrated proactivity, assiduousness and competency, rather than, for example, by means of one's demonstrated intelligence (e.g., the ability to come up with "good ideas"); prior social

connections (e.g., a personal relationship with influential volunteers who already earned their status in the group); or pre-established credentials (e.g., academic degrees or achieved ranks in the high-tech industry).

Conversely, if we follow the implications of dookratia for the social personification of PKW participants, we find that the group's system of governance is required by the psychosocial need of social persons for appreciation and approval, which, in this case, finds its expression in a political notion of justice according to which one's share in a common good approximates the amount of work that one has invested in the creation of that good. Interestingly, this local notion of fair distribution reflects the assumption of both Adam Smith and Karl Marx that the most elementary measurement of a good's value is the amount of work required for its creation, and the former's claim that all workers should be compensated proportionately, in accordance with their share in the division of labor.

And so, we not only find that PKW regime-type of dookratia confirms the Marxian claim that culture can be partly determined by economic interests; we also find, that ideological apparatuses of justification can be perfectly aligned with candid ideals of justice and fairness, which, in this case, parallel the ideals of a just society that Adam Smith advocated in his influential book The Wealth of Nations ([1776]2003).

Dookratia as a Form of Civic Action

Insofar as the style of la'asot cod and the event of terumat cod are the only communication practices of software production by which participants constitute the identities of assiduous, proactive and competent Python/Django web developers, these practices must also function, simultaneously, as means for dookratic participation in these groups.

Notably, those civic means preserve some of the original meanings of the term democracy in the ancient Greek sense, while actualizing the libertarian utopia of voluntaryism as a principle of self-governance that rejects any centralized form of authority "in theory and practice as incompatible with libertarian goals" and "seeks to delegitimize the cooperation and tacit consent on which state power ultimately depends" (Watner, Smith and McElroy 2017:3).[69] While it is obvious that the notion of dookratia in PKW volunteer setting is a twist on the composite of demos ("the people") and kratos ("power") that substitutes the noun demos with the verb do, it is less apparent that the group's preference for rule by the individual actions of "the 20% who make code" is semantically closer to the original Greek sense of the term

[69] While most of PKW participants are far from being libertarian revolutionaries in this anarchist sense, they nevertheless display a deep attachment to the principles of individual liberty and self-interested action in the organization's volunteer setting where they choose to work in their leisure time and for no monetary compensation. This attachment is clearly evident from the constitutive rights of voluntary participation and voluntary selection of tasks and their associated privileges of non-distraction and freedom from imposition.

"demokratia." In the explanation of political historian Josiah Ober (2008:3-5):

> In modernity, democracy is often construed as being concerned, in the first instance, with a voting rule for determining the will of the majority. The power of the people is thus the authority to decide matters by majority rule. [However], the original Greek meaning of "democracy" referred to "power" in the sense of "capacity to do things." "Majority rule" was an intentionally pejorative diminution, urged by democracy's Greek critics […] Demokratia, which emerged as a regime-type with the historical self-assertion of a demos in a moment of revolution, refers to a demos' collective capacity to do things in the public realm, to make things happen. If this is right, demokratia does not refer in the first instance to the demos' monopolistic control of pre-existing constitutional authority […] Rather it means, more capaciously, "the empowered demos" — it is the regime in which the demos gains a collective capacity to effect change in the public realm. And so it is not just a matter of control of a public realm but the collective strength and ability to act within that realm and, indeed, to reconstitute the public realm through bold action.

This definition of the term "demokratia" is compatible to some extent with the libertarian doctrine of voluntaryism. In both cases, individual actors are concerned in the first instance with promoting their personal agendas through the selection and execution of tasks rather than with a "voting rule for determining the will of the majority" (Ober 2008:3). This prioritization of practical action over voting holds true in PKW volunteer setting as one's code donations are immediately merged into his team's product if they meet certain technical requirements. Thus, within the framework of PKW eKnights, the common good is not a communal way of life but a software product, action is not public deliberation but an executable code, and the goal of the individual actor is not to mobilize others to do

something together, but rather to offer independent contributions to a collective enterprise.

Given that PKW BDFLs and their co-developers are the 20% who make 80% of their project teams' source codes, it is accepted without question that these developers' programming time worth more than that of most other participants (in accordance with the aforementioned Smithian principle of fair distribution). This social agreement is clearly apparent from the division of labor between code donors qua 'beneficiaries' and BDFLs qua 'benefactors' in the event of "terumat cod." To demonstrate this, let us return to excerpt 17, from p. 153:

1. I'm very much in the business of show me a code.
2. […] Many times someone can come and say, I think that the map should be red and the blocks should be yellow. Okay ((chuckles)) […] it will be easier if you do some kind of example, a version of it, of a map in red and the blocks in yellow, let me see it and if it looks better, then obviously it will get in.
3. […] By the simple fact that you did something good, you contributed to the project in seconds.

The original analysis of this excerpt has focused attention on the material potentialities of computer code for simplicity, literalness and immediacy to argue that any rational BDFL will find it more efficient (and thus preferable) to "see" a feature like "a map in red" than to imagine how this feature might have looked like if the person who tells him about it would have actually taken the time necessary for making it. Clearly, the requirement from volunteers to "show code" allows BDFLs to spend less time on speculative discussions with people who might or might not donate "good code" to their repositories. However, it

is equally clear that this requirement comes at the expense of individual volunteers who now need to take the economic risk of code rejection every time they want to donate code. In this way, the regime-type of dookratia provides a measure of institutional protection to the most productive volunteers.

Conversely, the BDFLs who receive this privilege are expected to dictate the directions of product development 'dookratically,' by virtue of their prevailing assiduousness, proactivity and competence. This requirement for leadership by example is expressed in the following commentary:

(42) Interview (10/1/2016)

[The BDFL] is a very significant role and I really feel that at a given moment the amount of time I invest is the upper limit of the investment of all the others. […] And then, if I'm in a period of lower investment then there will not be anyone who will invest at all. And if [my] investment is high then it allows people to stay somewhere in the range under it.

Nevertheless, as Yoav insists below, a BDFL's commitment to "dookratia" entails a deference of his ownership right in the unlikely event that one of his peers claims the reigns of project leadership by his individual merits:

(43) Interview (10/1/2016)

1. Yoa: Look […] it's not that important for me to be a dictator for that matter.
2. In general yes, I dictate the directions that [the team] should follow. But if someone will come and say I have a different direction, so it will be accepted if he actualizes that direction. Like you have to actualize it ((smiles)). And it's not enough that you'll say I have a different direction and it is better.
3. Nim: So like the burden of proof is on you.
4. Yoa: Totally. Yes.

Finally, like any system of governance that oversees a concrete regime of work, "dookratia" relies on expressible and

followable sets of rules. The nature of these protocols and the ways in which they are used to embody the group's valued model of personhood are elaborated in the following section.

The Rule System of Dookratia

As chapter 7 discussed, the SFCPs "terumat cod" (code donation) and "la'asot cod" (to do/make code) are the primary means of expression by which participants in PKW community of practice fashion the presentation of their public images as assiduous, proactive and competent programmers. At the same time, the present analysis has shown that these two practices are also the primary means of expression by which participants in this community of practice can constitute a dookratic polity.

If both of these findings are correct, then the analysis of "la'asot cod" and "terumat cod" as elements in PKW discourse of work's rule system conducted here should bring the thesis of this study to its conclusion as it will allow us to demonstrate that the basic economic interest of an individual rational actor maximizing utility may lead to the communicative constitution of a full-fledged organization of production (i.e., a composite of a technosocial system of product development and a system of governance). To use the same graphic schema from chapters 6 and 7, this relationship can be expressed as so:

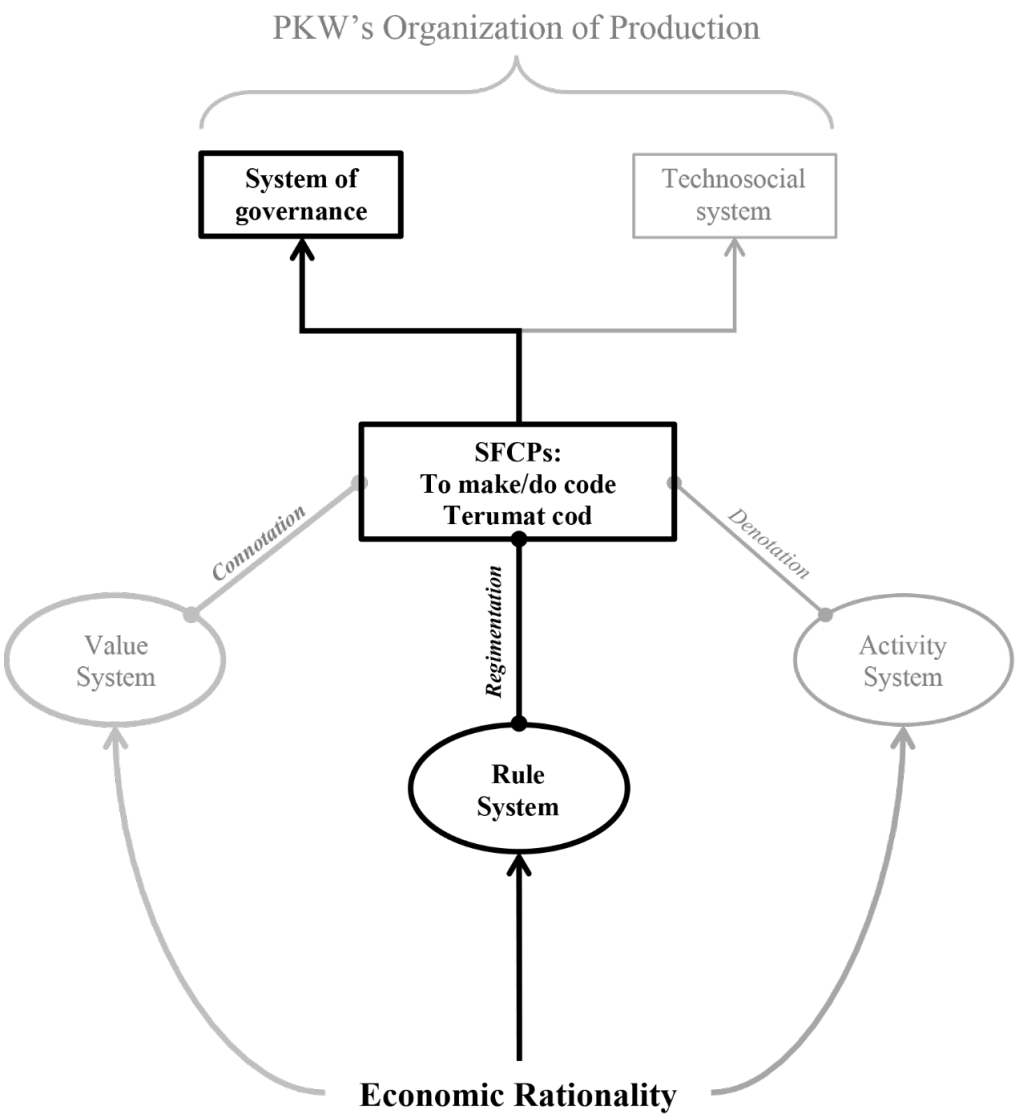


Fig. 15. The Thesis of Chapter 8

The examination of "la'asot cod" and "terumat cod" in their capacities as elements in PKW discourse of work's rule system is designed to tie between the analysis of this discourse of work's activity system in chapters 6 and the analysis of its value system in chapter 7. That is, the forthcoming analysis shows a full causal relationship between the economic rationality of efficiency optimization (under the conditions of time scarcity

and lack of command ability) and the organization of production components of technosocial system and system of governance, while stressing the essential role of local means and meanings of communication in its actualization.

Analytical Framework and Procedure

The analysis of "la'asot cod" and "terumat cod" as elements in PKW discourse of work's rule system follows from Carbaugh's (1990) framework for the formulation of *code rules* and *normative rules*. In this perspective, code rules are defined as norms for interpretation (cf., Hymes 1972a) that specify how certain features of a communication practice are linked to the discursive hubs of personhood, activity and social relations. The central function of such rules is therefore to create "mutual intelligibility and shared coherence in communicative action" (Carbaugh 1990:140). Analytically, code rules are stated in the form: "in context C, the unit X, counts as meaningful on another level as y, y´…" (ibid).

In contrast with code rules, normative rules are "abstraction of patterns for acting" that bring into focus the sequential organization of social conduct. Such rules "derive from pre-existing templates and provide standards for judging what to do, and for evaluating whether what has been done, has been done properly" (ibid:141). As abstract rules for acting appropriately, normative rules are stated in the form: "in context C, if X, one should/not do y, y´…" (ibid:142).

While code rules and normative rules are phenomenally and analytically distinct, they converge to a great extent in the case of PKW style of "la'asot cod." The reason is that volunteers in the organization's eKnights are not expected to do much more (or less) than "showing the code" they write to their BDFLs through the act of "pull-request" that opens the online event of "terumat cod." The following analysis therefore starts with an attempt to formulate the basic normative and code rules proper to the style of "la'asot cod," and then specify the normative rules that govern the communication event of "terumat cod."

Rules of La'asot Cod

The normative and code rules that structure the communication style of "la'asot cod" as a primary means for civic participation in PKW 'dookratic polity' can be classified into three protocols. The first two protocols specify normative rules that derive directly from the libertarian contract of voluntary participation and voluntary selection of tasks and its underlying condition of lack of command ability. The third protocol specifies code rules for the correct performance of PKW valued model of personhood. The analysis uses a legalistic style to emulate the formality of the rules. Accordingly, the instructions listed below could be treated, by the reader, as rules that if followed would help him to successfully integrate into any one of PKW eKnights if he would be a sufficiently competent Django/Python web developer.

I. NORMATIVE RULES OF VOLUNTARYISM. Any individual who wants to participate in PKW community of practice is required to abide by the contract of voluntary participation and voluntary selection of tasks, which prescribes the following prohibitions. First, the participant must not expect or ask anyone else to perform any kind of work regardless of the ingenuity or significance of his ideas. Second, the participant must not ask the BDFL of his eKnight of interest for permission to perform programming tasks with the expectation that such a permission would guarantee the acceptance of his intended contributions into that eKnight.

II. NORMATIVE RULES OF WORK EFFICIENCY AND TASK EXECUTION. In the context of PKW community of practice, whenever a volunteer self-selects a task, he should perform that task in an efficient and timely fashion by "sitting and working" or "making code." While coding/programming is always prioritized over speaking, the volunteer may propose his intended course of action to the BDFL if he wishes to do so, and may also pose the BDFL with specific technical questions about the project's source code and/or development environment. However, as Shimon emphasizes in the following excerpt, these questions must meet the requirements of concision and relevancy:

(44) Interview (10/1/2016)

It's not that I do not want to be asked questions. I totally want to help. But the questions must be relevant. You have to see […] that one asks you of something that he really couldn't understand alone.

In support of the stated thesis of this study, we see that the two protocols specified above are caused by and provide an

expression to the economic interest of efficiency optimization. First, by requiring participants to minimize their speech activities, these protocols help to increase the productivity of the group's BDFLs while allowing them to enjoy their libertarian rights and privileges of non-distraction. Second, by instructing participants to avoid "prolonged discussions," the rules help to save the time that entire teams would otherwise spend on disagreements and conflict resolution.

Third, by requiring participants to "establish facts on the ground" through the performance of unidirectional acts of creation, these normative rules ensure that the production process will proceed with a minimal interruption, as potential disagreements are effectively prevented by the fact that one "does something useful" while his potential antagonists do not. Moreover, given that participants have no guarantee that their donations will be accepted to the products, they are expected to exercise their voluntary will in a determined and bold manner.[70] This expectation leads directly to the third protocol that specifies additional rules for the correct performance of a 'virtuous personhood' in this particular volunteer setting.

III. CODE RULES FOR THE PERFORMANCE OF A VOLUNTEER SELF. The relative ability of a volunteer in PKW community of practice to

[70] In this regard, more calculated volunteers may attempt to manage their risks based on the evident fact that the simpler one's contribution is and the more it is aligned with the development directions dictated by his BDFL, the more likely it is to be merged into the product.

display the traits of an assiduous, proactive and competent programmer becomes apparent only at the point where he opens the communication event of code donation with the act of "pull request." That is, a volunteer may self-select a challenging task, ask the relevant BDFL a few concise questions about the eKnight's source code, and then present an impression of himself as executing that task (e.g., by silently sitting in front of his computer screen within the framework of a development meeting). While this course of action is certainly favored by the group's core teams as it does not "make noise," it has no intrinsic value to these participants' technosocial system of product development. The value of a potential volunteer to an eKnight, then, can only be assessed at the point of online contact when this individual sends a "pull request" to that eKnight's BDFL.

The two central code rules that govern this social evaluation of one's volunteering self can therefore be formulated as follows: (i) in the context of PKW community of practice, a volunteer who "shows the code that he sat and wrote," counts as a proactive and assiduous person who understands the principles of dookratia; and (ii) if the code thus shown solves a product issue or implements a new feature or modification in the best possible way, the volunteer may also count as a competent (or even "superb") Django/Python web developer.

While these rules are valid in a general sense, the worth of a volunteer who displays the virtues of proactivity, assiduousness and competency is ultimately determined by the

consistency of this presentation of self and the amount of contributions that such a person makes over time. The more committed the volunteer, the more valuable he becomes to his team. Volunteers who display the virtue of competency but keep their commitments down to a minimum are thus recognized as valuable yet unreliable code donors. This normative assessment may prevent BDFLs from letting such individuals implement modifications whose maintenance relies on their particular areas of expertise.

Participants who keep asking for guidance or permission to perform their tasks with the expectation that the group's BDFLs will show recognition and appreciation of their investment of time, efforts and good intensions may be stigmatized as "trolls," or (in this specific social scene) as reactive persons who do not understand or have insufficient self-reliance, determination, or confidence to follow the protocols of voluntary participation and voluntary selection of tasks.

In actuality, such "trolls" have small value for their project teams even if they excel in the virtues of assiduousness and competency, because they: (i) violate the group's BDFLs right for voluntary, autonomous and self-interested action; (ii) disturb the balance between the BDFLs' privilege for code rejection and the code donors' privilege for unaccountable exit from an OSS project; and (iii) require an investment of time that reduces the BDFLs' productivity and may not pay off due to their questionable commitments.

The following discussion completes the analysis of "terumat cod" presented in chapter 6 by examining the contextual constraint of Norms in Hymes's (1972a) SPEAKING acronym. The reason for this delay is the reliance of PKW discourse of work's rule system on the valued model of personhood interpreted in the previous chapter. In the interest of simplicity, the analysis presented below is focused on the following instance of "terumat cod" whose interactional terseness and keying of immediacy are heard loud and clear.

(44) GH_E02

Feb 11, 2016

100. Topic: Refactored css definitions for a less file
101. Ron: @Tom Does that make better sense this way?
102. ((list of code contributions))

Feb 14, 2016

103. Tom: yes, and also you need to add an import to the [less file] here:
https://github.com/hasadna/Open-Knesset/blob/master/less/app.less

Feb 16, 2016

104. Tom: @Ron?
105. Ron: Sorry, missed the mention.
Doing it right away.
106. ((list of code contributions))
107. Done.
@Tom now it's ready.

Feb 17, 2016

108. Tom: merged commit d78032c into hasadna:master

Given that the communication event of "terumat cod" — at least as an ideal type — is embedded in the situation of a development meeting whose relatively high degree of organization

has been defined in terms of Irvine's (1979) framework for the study of formality, it will be useful to think through the same analytical lens here. In this approach, the formality of a communication event is defined in relation to four analytical continua: (i) the continuum of *situational focus* whose poles are task-oriented and spontaneous activities; (ii) the continuum of *positionality* whose poles are structural and personal identities; (iii) the continuum of *code consistency* whose poles are reiterative and improvised performances; and (iv) the continuum of *code structuring* whose poles are fixed and tentative rules for the selection and usage of communication channels (e.g., the degree to which a specific dressing code is formalized and/or mandatory in a particular social scene).

In terms of these continua, the principal normative rule for situational focus and positionality that dictates the formality of "terumat cod" can be stated as follows. In the context of PKW community of practice, whenever volunteers engage in product development via the communication event of "terumat cod," they should focus only on the technical tasks at hand, i.e., showing, reviewing, revising and accepting/rejecting code contributions. The volunteer who uploads his contribution to a project's GitHub subdomain should therefore function only as a "code donor" and the BDFL and/or co-maintainers of that project should function only as "code reviewers." This principal rule is complemented by the following set of normative rules for code consistency (in Irvine's sense).

First, whenever a code donor uploads a contribution to an eKnight's GitHub subdomain, he must fill the topic section at the top of the pull-request page. This text must provide a general description of what the donated code does in concise technical language (e.g., "Refactored css definitions for a less file," L44:100). The second normative rule of code consistency states that the "code donor" may, but does not have to, add a comment in the box that shows all the modifications and/or additions that he has made to the project's source code. Ideally, this text should be written in a descriptive form and specify issues of technical concern in the most succinct terms possible (e.g., a technical description of a feature's limitation). However, this is not a strict rule, and the "code donor" may perform at that place other actions as well. For example, in L44:101, the "code donor" Ron asks the "code reviewer" Tom a yes/no question to express an uncertainty about the necessity of his proposed modification. That said, comments in this box should refer only to the lines of code that appear below them.

The third normative rule for code consistency in Irvine's sense states that the "code reviewer" should respond to the "code donor" only after he has reviewed the contribution. If the code does not require any revision, the "code reviewer" may merge it into the product without further delay. If the code requires a revision, the "code reviewer" should ask for it in a concise technical language. If the request requires an accommodation to the particular technical standards of the project, the "code

reviewer" should provide the "code donor" with the necessary information, usually via hyperlinks as in L44:103. If the rationale for the request is not self-evident, then the "code reviewer" should also provide further explanation for why the revision is necessary (as in L18:2, p. 159).

The fourth and last normative rule of code consistency states that whenever a "code reviewer" requests a revision, the "code donor" should respond only one time with the already revised code, rather than two times, first to accept/reject the revision request (as speakers usually do in face-to-face interactions), and second to submit the revision itself. That is, the code donor should do the necessary work and upload the revised code upon its completion with or without a new commentary that confirms the revision and/or specifies further technical issues of concern.

The time interval between the point at which the request is made and the point at which the revised code is uploaded to GitHub defines the "code donor's" proper conduct. If the "code donor" does not respond in a timely manner (that may be defined in relation to the complexity of the requested tasks), the "code reviewer" will check on his work. In response, the "code donor" should upload the revised code without delay. The above excerpt provides a concrete example of this process. In this case, Ron does not see that Tom requested a small revision. This leads Tom to contact Ron after two days with the phatic question "@Ron?" (L44:103). When Ron sees this question, he immediately turns to

perform the revision and then uploads the new code with an apology for missing Tom's request (L44:105-107). This apologetic move provides evidence that both Tom and Ron are aware of the possibility of a rule violation. Tom through a reading of Ron's silence as possible neglect, and Ron through an acknowledgement that he violated the rule of timely response.

Taken together, these rules and dimensions of formality structure the event of "terumat code" as an instrumental occasion that leaves very little room to societal interchanges such as deliberative discussions or arguments. Thus, for example, one could not guess that Ron (the BDFL of an eKnight) and Tom (a co-maintainer in the eKnight to which Ron offers a code contribution) know each other rather well from the group's development meetings. It is precisely in this sense that the communication style of "la'asot cod" as a mode of dookratic participation governs the structuration of "terumat cod" as PKW central public event of decision making or governing.

Normative Obligations and Rights

In addition to the rule templates specified above, "code donors" and "code reviewers" in the event of "terumat cod" are expected to respect the following obligations and rights.

I. It is preferred that "code reviewers" will acknowledge "code donations" in an efficient and timely fashion that replicates the proper conduct expected from any "code donor" who "sits and work" rather than "make noise." Given that much of the

OSS production process depends on the availability of BDFLs and their co-developers,[71] the economic logic of efficiency optimization underlying this normative obligation is apparent from the fact that if the central figures in the production process will fail to show a degree of commitment that exceeds the one displayed by the most assiduous, proactive and competent participants in their teams, they are likely to experience a decrease in the overall amount of contributions to their source code, a decrease that could lead to that source code's demise (see also excerpt 42 above).

II. A "code reviewer" has the right to ask a "code donor" to revise his "code donation" in any way that that reviewer sees fit. This right to condition the acceptance of code donations by the correct performance of additional work from "code donors" derives from the extra responsibilities of "code reviewers" to the overall quality of the projects' source codes.

III. Any "code reviewer" has the ultimate right to reject code donations by his autocratic ownership right. While the exercise of this right is relatively rare, when it does occur, the BDFL must provide the code donor with a technical account for the decision but he is not obliged to sweeten the blow with societal forms of courtesy. Danny explains this obligation for accountability and the right for using a direct, bold-on-record mode of communication (Brown and Levinson 1987) as follows:

(45) Interview (1/12/2016)

[71] See also Weber (2004:116-117).

1. Dan: If we're talking about someone who [uploaded] a particular [code contribution], and the [BDFL] decided that he does not accept this [contribution], he'd write him in the reject- he'd write him why.
2. Nim: Does it make sense that a [BDFL], say, will filter out [a code contribution] without giving an account?
3. Dan: This is not happening. It does not happen. It does not happen. The account may be given in two words. Lame code, for that matter. I'm not quoting anyone, okay? But no one here is obligated to be nice. We all work here for the same cause and eventually ((laughs)) most of the programmers here would not pass as hosts in the [TV] kids' channel. This is in terms of how positive and smiling they usually are.
4. Many times when there is something to talk about and when someone really made a certain mistake or there is something else that needs to be checked […] then there will be people on the team who will direct you to where your mistake was, what you need to learn more, [or to] better ways to achieve the same result.

Under the dookratic regime of "la'asot cod," BDFLs may account for source code rejections in a manner that could be considered as rude or impolite in other social and interactional contexts. Nevertheless, the account itself is mandatory — a normative requirement that provides us with clear evidence that the BDFL qua autocrat is constitutionally subject to the rule system of "dookratia."

IV. "code donors" who want their code donations to be merged into any OSS product under their own authorship are obliged to own their self-selected tasks and thus to write and revise their code donations (if asked to do so) without any formal assistance from the reviewers. This requirement for code donation ownership parallels any BDFL's right of project ownership. The difference between the two rights is that BDFLs are also recognized as the legitimate owners of the piece of noosphere or the 'frontier land' of civic and programming thought

on which the project was first conceived and built. This legalistic arrangement of ownership rights is designed especially to mitigate the autocratic power of the BDFL's right of code rejection as Itay explains below:

(46) Interview (7/1/2016)

1. I do not remember getting to the point where I'm telling someone do it contrary to your opinion. Because he will not do it anyway, he is a volunteer.
2. And he is a benevolent dictator on the little plot of code.
3. Let's say that it's much more like a feudal model than a modern dictatorship if you will. Like there is a division of powers, it is not equal but there is also a lot of decentralization.

Here, Itay is commenting on the group's dookratic system of governance as a whole in two interrelated ways. First, Itay demonstrates his understanding that his autocratic authority is essentially limited by the condition of lack of command ability as any volunteer has the contractual right of voluntary participation and can therefore perform an unaccountable exit from any OSS project. Secondly, Itay qua project owner uses notions such as "decentralization" and "feudalism" to affirm that the absolute power of an autocrat is neither possible nor desirable in PKW volunteer setting.

Coda

In excerpt 46 above, Itay in effect brings this dissertation to its empirical conclusion. It is now shown that the economic principle of utility maximizations leads individual rational actors to constitute culturally specific values and organizational norms, the terms for these constructs, and the

rule-governed activities on which they depend for their intelligibility. This finding, and its implications for the fields of Cultural Discourse Analysis (CuDA) and microeconomics, are elaborated in the following, concluding chapter.

CHAPTER 9

CONCLUSION

Introductory Remarks

The two criterions that I see as essential for the evaluation of any social scientific study are the following. First, a proper or 'good' study is one that advances a single, transparent and concise thesis. Second, a good study is one that uses empirical data to present its thesis in the form of an irrefutable finding. Based on these criterions, the present study has attempted to defend the thesis that:

> The emergence of some socially constitutive communication practices can be best explained by constant human nature factors such as, for example, the factor of rational action/choice in the case of PKW organization of production.

The following discussion explicates this thesis in relation to the empirical findings specified in the data-based chapters, and draws out its implications for communication and microeconomic inquiry. The conceptual framework used for this purpose is Carbaugh and Hastings's (1992) distinction among three modes of theorization that occur simultaneously in any CuDA research, albeit in different degrees of salience. The first mode, *theorization of the case*, involves a description and interpretation of communication practice as it is recognized and used by members of a given community or group. The second mode, *activity theorization*, draws attention to classes of communication practice and is therefore of a syntactic, broad

type, rather than of a specific sociocultural domain. By using it, CuDA scholars develop descriptive frameworks that inform ethnographers how to identify and explain possible variations in what they might see or hear across social settings. The mode of activity theorization thus grounds specific analyses and lays a base for comparative and cross-cultural analyses.

The third mode of theorization, which Carbaugh and Hastings call *basic orientation*, concerns the paradigms of entire disciplines and areas of research. This mode of theorization requires the acceptance of certain epistemological and ontological assumptions about the nature of the world and the human place within it. As such, it concerns the conceptual horizons against which specific activity theories and the range of empirical phenomena they help classify achieve a minimal level of intelligibility vis-a-vis the ethnographers who engage them.

Drawing on this tripartite framework, the following discussion attempts to reconstruct the above stated thesis in a 'bottom-up' fashion that replicates, to an extent, the actual procedures that led to its original formulation.

Theorization of the Case

The practical question that triggered and guided this study can be restated as follows:

> What are the communication means and meanings by which participants in PKW eKnights manage to create and optimize the efficiency of a voluntary organization of production under extreme conditions of time scarcity and lack of command ability?

Before presenting the empirical answer to this question, it is important to stress that the question, as formulated above, brings together the assumptive bases of microeconomics and CuDA. On the one hand, the question presupposes that PKW participants are rational actors maximizing utility who move and operate under an existential condition of scarcity, and on the other, that such economic rationality depends on local means of expression for its intelligibility and efficacy.

My attempted synthesis between these two assumptions and its broad theoretical implications for CuDA and microeconomic theory are elaborated in the following section. In the present section, these assumptions are stated explicitly to remind the reader of the hermeneutical logic of social scientific discovery. As Western philosophers from Plato to Heidegger have shown, a question can only make sense within the paradigm (or basic orientation) that the person who asks it presupposes. The acknowledgement of this condition, which appears to be an ontological constraint upon human understanding in general, allows us to bring together apparently disparate epistemologies in the social sciences. Within the framework of the present study, one could say that the 'positivistic' questions and hypotheses of conventional microeconomics can only make sense under the essentialist supposition of rational action/choice (in either its metaphysical or naturalistic registers). Similarly, one could say that CuDA researchers must presuppose a descriptive taxonomy of social units and the belief that such units are made

by the people who use them, in order for their ethnographic explorations to make sense.

Summary of Analysis Results

This study solves the empirical puzzle of PKW organization of production with the observation that participants in the group's eKnights use a local discourse of work whose primary Symbols For Communication Practice (SFCPs) are the style of "la'asot cod" (to do/make cod) and the event of "terumat cod" (code donation) that this style helps organize. The activity, value and rule systems of this discourse of work enable group members to constitute: (i) a technosocial system of Benevolent Dictatorship For Life (BDFL); and (ii) a dookratic system of governance that embodies personhood values of assiduousness, proactivity and competency with rule templates for autonomous, independent and self-reliant action.

The technosocial system of BDFL provides eKnight members with concrete means to overcome practical problems of coordination and scaling (i.e., how to put together a functioning piece of software in the desired context of growing numbers of participants, and under the limiting condition of lack of command ability). Similarly, "dookratia" as a system of governance provides eKnight participants (and especially BDFLs) with means to mitigate the economic risk of workforce recruitment (i.e., how to identify productive and committed 'persons of virtue' among any number of newcomers to a development meeting), and to solve

the political dilemma of project governance (i.e., how to make decisions and resolve conflicts within a highly distributed production process).

Generalization of the Case

By way of generalization, one could say that the discourse of work through which PKW participants constitute an efficient organization of production depends on two culturally sanctioned unquestionables. The first unquestionable is the assumption that one's original creations are one's private property in the Anglo-American legalistic sense of property as *the right to exclude*. To question the right of a project founder to exclude others from his production process by means of ownership claims is to undermine the communication event of code donation and the BDFL institution it helps constitute.

The second, related, culturally sanctioned unquestionable is the assumption that people desire (or should desire) to exist as self-sufficient individuals who do what they want with minimal reliance on others due to owning or acquiring personhood traits of assiduousness, proactivity and (technical) competency. To question the attractiveness or desirability of this libertarian model of personhood from, say, a communist viewpoint, is to undermine the communication style of "la'asot cod" as a dookratic mode of civic participation and the Smithian notion of fairness it helps reproduce. In PKW community of practice, the amount of "good" or "useful" code one writes defines the degree to which

one influences the development of his group's product. Arguing that all the participants in an eKnights should have equal rights of decisionmaking regardless of how much code they donated would be nonsensical in this particular volunteer setting.

Similarly, the moral and political argument that a BDFL has an inherent responsibility to accommodate a potential volunteer's physical or socially structured disability to contribute "good" or "useful" code entirely on his or her own has no resonance in the group's dookratic polity.

So What?

The analytical move from a practical question to its context-specific solution and the development of that solution as a theory of the case in Carbaugh and Hastings's (1992) sense are sufficient for the production of a full-fledged ethnographic report. Such a report would certainly have value for participants in the group under study as it would provide them with reflexive means to think about their taken-for-granted knowledge and practices in conceptually formal and analytically precise terms. Such a report might also have some practical value for participants in similar voluntary corporations who are faced by the same obstacles of social cooperation.

However, an ethnographic study that results only in such a theory of the case remains limited in two substantial ways. First, a theory of the case cannot be generalized, at least not in formal-analytical terms, in the absence of an activity theory

that abstracts the localized practices it models in context-general terms. Second, a theory of the case has very little significance for the social scientific community, and especially for scholars working in different areas of research or with different sets of data. Albeit the empirical puzzle that such a theory solves may attract the attention of curious readers and observers, those readers will not learn from it something of a paradigmatic significance. Hence the proverbial 'so what' question in the defense of a socially scientific thesis. My answer to this question, and thus the defense of this dissertation proper, is elaborated in the following two sections.

Basic Orientation

Why should anyone who is not a professional programmer care about the finding of this dissertation? An explicit and reflexive usage of the hermeneutical circle described above leads us to find the answer to this question in the basic orientation that the empirical puzzle of PKW organization of production and the theory of the case that solves it must presuppose as a condition for their intelligibility.

As discussed in the previous section, in order for PKW puzzle to make sense, one must presuppose that the participants in the group's eKnights are rational actors maximizing utility and that the economic rationality of these individuals can only be expressed through locally recognized Symbols For Communication

Practice (SFCPs) and the activity, value and rule systems in which these SFCPs are embedded.

Insofar as the theory of the case presented above cannot be rejected or replaced, the evaluative question becomes: what can one learn from the fact that the veracity of these presuppositions must be affirmed in order for one to understand how and why PKW organization of production works? To address this question, the study has conceptualized the group's organization of production as a 'response variable,' the economic rationality of the individuals who operate it as an 'explanatory variable,' and the discourse of work that articulates and actualizes this rationality in the constitution of that organization of production as a 'mediating variable.' The study's analytical thesis thus became that social organization tracks economic rationality in voluntary corporations such as PKW eKnights, and that this causal relationship is actualized through the creation and usage of local SFCPs. I believe that this formulation of the study's thesis should be of interest to any social scientist who takes issue with the relationship between economic rationality and social organization and/or with the role of communication studies in the development of a cross-disciplinary paradigm whose topos is the human condition.

Analytical Significance for CuDA

The main analytical implication of this thesis to theory building within CuDA can be stated as the following proposition.

If there are social and psychological human nature factors; and if these factors place observable constraints upon the communicative actions through which they are articulated and actualized; then such factors principally fall within the purview of CuDA scholarship and should, therefore, be incorporated into its descriptive and interpretive framework.

The theory of the case that solves the empirical puzzle of PKW organization of production provides a direct evidence in support of this proposition as it shows how one such human nature factor (i.e., rational action/choice) requires expressive means of communication for its efficacy while directing and shaping these very means of expression.

Given that the same can be said about Goffman's theory of the social self (Rawls 1987), and that this theory articulates elements from Durkheim's social psychology with ethological and evolutionary approaches to the study of other animal species, one could expect to find that rational action is not the only human nature factor that one needs to consider when trying to explain the emergence of a given cultural system of communication.

Analytically, then, there is a role for human nature factors in the description and interpretation of communication practice. Given that this role is yet to be defined, the present study opens up a new and exciting research area within CuDA. Some elementary questions that could guide that line of research are:

- What are the human nature factors that place constraints upon social interaction, and how such factors can be best documented and described?

- Are there specific domains of social life or classes of social and economic practice that reserve specific roles for such factors?
- How do such factors relate to each other in and through the communication practices they shape and on which they rely for their expression?

Theoretical Significance for CuDA

While human nature factors such as rational action/choice may be essential to the description of some communication practices within some sociocultural contexts, their real significance for CuDA is their ability to constitute a precise theory of human agency. Such a theory could enrich and deepen ethnographic explanations of communication practices and the social structures that these practices help constitute. With a measure of caution, one could say that at least some CuDA explorations had to presuppose the veracity of at least some human nature factors in order to give better sense to their empirical findings. Integrating these factors to CuDA theory would therefore help systematize some of the covert knowledge that ethnographers of communication have already been using.

The development of explanatory frameworks on the bases of human nature factors would also serve to mitigate the intellectual tendency of CuDA ethnographers to conceptualize the individual human as a socially constructed and culturally variable being (Shweder and Bourne 1984). Insofar as human agency must be external to any sociocultural and communication system by logical necessity, such an agency, and the role it might play in

the organization of social (inter)action can only be defined in terms of constant human nature factors. This should be of particular interest to CuDA scholars whose work centers on problems of social, cultural and ethnolinguistic identity.[72] While the once provocative claim that human identity is a product of social interaction can now be considered as trivial, ethnographers have largely forgotten that the economic needs of a rational actor or the psychological needs of social persons for acceptance and respect (in Goffman's sense) play a crucial role in shaping the practices through which they are articulated with culturally variable notions of personhood.

Between CuDA and Microeconomics

While this study is addressed primarily to CuDA and communication scholars, it offers analytical and theoretical contributions to microeconomic and utilitarian research. Analytically, it proposes that if rational action/choice depends upon culturally specific means of expression for its articulation and efficacy, it can and should be studied ethnographically within the natural settings in which it occurs. As explained in chapter 2, this is not to challenge the merits of conventional microeconomic research in an 'either/or' fashion, but rather to suggest that microeconomists could benefit from the specializing

[72] As influential anthropologist Sherry Ortner (1984) has shown, these issues are essential to the contemporary American society where processes of diversification and struggles among more and less powerful groups are popularly conceived in terms of identity politics.

tools developed within communication studies, as such tools may be fit to address economic phenomena that they have so far been unable to see or explain.

After the dookratic model of PKW participants, this study not only claims that microeconomists could benefit from ethnography of communication research tools. It also provides a 'proof of concept' for this claim, as well as a precise and systematic way for its operationalization. In this respect, one can see a role for CuDA in the application of economic theory to real-world problems of social cooperation. To the extent that CuDA ethnographers produce applicable models of the activities they study together with 'cultural guides' for the interpretation of these activities, they may not only contribute to the scholarly understanding of human decisionmaking (or choice) but also to the fashioning and improvement of economic practices of production and exchange among participants in both voluntary and commercial corporations. This, I think, is where CuDA may be most useful to corporate personnel whose abstract economic models miss the reliance of actors on local means of expression, means that require the orienting and moralizing function of societal values and norms (Streeter 2011). As the analysis developed in this dissertation shows, the creation and optimization of a voluntary corporation's organization of production depends on culturally specific unquestionables which are not, themselves, rational.

On an explanatory level, this study suggests several directions for the development of political economic theory in

both of its utilitarian and Marxian registers. First, as we have seen, the relationship between economic interest and its symbolic and representational means of expression is variable and complex. While the private and group interests of individual actors may be articulated with apparently unrelated symbols and meanings as Marx suggested, such articulations are not necessarily aimed at the creation of false consciousness. To the contrary, a distorted representation of a group specific 'class interest' (such as that of PKW project owners/BDFLs) may reflect commonly shared and philosophically defensible notions of fairness and social justice.

At the same time, what is true on a microsocial level (such as that of an eKnight) may not hold on a macrosocial level (such as that of capitalist market economy). For example, while the Smithian notion of fairness may work well among participants in a voluntary corporation, the representational models through which it is articulated may be used to manipulate workers on a different level, i.e., persuade people to work harder for no remuneration or to increase the efficiency of commercial firms on the expense of one's well-being. Such models may also be used to promote competitive agendas that leave structurally disadvantaged people to fend for themselves within the industrial economy. Regardless of the politics surrounding this discussion, it opens a new direction for economic explanation as it brings together the highly specialized micro-analytical tools of CuDA and the macro-analytical scopes of utilitarian and Marxist sociology.

A second direction for future research and basic theorization derives from the fact that microeconomists have not yet presented a theory of value capable of accounting for different kinds of utility and the relationships between them. Here, Goffman's (1967) distinction between substance and ceremony comes to the fore. For the economist, substantial wants such as one's want of shelter and ceremonial wants such as one's want to be treated with deference in a particular way within a given social setting are similar in that they can both be conceptualized as subjective desires whose satisfaction requires the expenditure of scarce resources. While this may be true, one needs to keep in mind that there are biological needs common to all members of the human species, and, conversely, that there are culturally specific needs that cannot be conceived or expressed outside of language and society.

In his analysis of the psychological makeup of the individual human being, philosopher Jean Jacques Rousseau ([1775]1987) famously distinguished between two types of self-interest. The first type, *amour de soi* ("love of oneself") is "a natural sentiment which moves every animal to be vigilant in its own self-preservation" (ibid:106). The second type of self-interest, *amour propre* ("vanity"), presupposes social ideas about the accumulation of property, wealth and status that require comparisons between self and others. Such ideas and comparisons may incite negative sentiments (e.g., jealousy, humiliation, resentment and hatred) that could give rise to competitive and

violent desires that have no place in the state of nature (e.g., the desire to outdo others or the desire for revenge).

While Rousseau used this distinction in his critique of European modernization, CuDA ethnographers and microeconomists may mobilize it for their own specializing purposes. For example, the reader of this dissertation could ask:

- How, if at all, does a presentation of one's self as an assiduous, proactive and competent programmer may allow one to gain and/or accumulate symbolic value within PKW volunteer setting?
- To what extent do such symbolic utilities motivate programmers to create voluntary corporations?
- How do such ceremonial motivations compare with more substantive motivations such as a programmer's desire to overcome a specific technical or bureaucratic obstacle that disturbs him at a material level?

While such questions are not entirely new, I believe that we do not yet have a proper way to address them. In fact, the very distinction between substantive and ceremonial utilities or motivations presents CuDA scholarship with a serious challenge, one that Rousseau acknowledged but not entirely solved. Insofar as this distinction varies as it can only be made locally by members of a specific community or group, how can one speak of truly biological or substantive needs that principally precede language and society?

Activity Theorization

The possibility for shaping communication practices through the substantial and ceremonial needs of a rational actor leads to

a further examination of the activity theory that arises from the practices of "la'asot cod" (to do/make code) and "terumat cod" (code donation). To the extent that these practices of and for efficiency optimization reflect a desire to substitute rhetorical, deliberative and conversational interaction with activities of software production, they speak directly to the performative class of communication acts that Searle called "declaratives" whose defining characteristic is:

> that the successful performance of one of [their] members brings about the correspondence between the propositional content and reality, successful performance guarantees that the propositional content corresponds to the world: if I successfully perform the act of appointing you chairman, then you are chairman; if I successfully perform the act of nominating you as candidate, then you are a candidate; if I successfully perform the act of declaring a state of war, then war is on; if I successfully perform the act of marrying you, then you are married.

While the coded instructions that programmers write fall within this sociopragmatic class of declaratives, they differ from the speech acts in Searle's formulation in one important respect. The declarative acts of programming by which participants in OSS communities of practice constitute a dookratic polity are durable, aggregable and negotiated. And to the extent that such political acts of creation have no mode of addressing an interlocutor, they constitute witnesses rather than publics in the traditional political sense.

To demonstrate this once more, let us consider the following excerpt in which Elihav recalls how Ilya, the creator of PKW second project, joined the original team of Open Knesset:

(47) Interview (1/30/2017)

1. So first of all we tried to make it very democratic because it sounded appropriate to the nature and purpose of the project. But we realized very quickly that it doesn't work. Like, things need to be done. And like people- once there is a line, people understand or see the agenda from the operations that happen.
2. I remember that we even had discussions about it. At some point we invented a name for it, what we called dookratia from the word la'asot [to do].
3. […] The most concrete example for this is how I met Ilya. […] One day I got a pull request […] that said, here, I made an RSS feed for the website. […] I reviewed the code, wow! Cool. OK excellent. We accepted, and two days later it was already in the product.

Elihav's notion of "dookratia" (L47:1-2) reflects the group BDFLs' assumption that political action involves the freedom of individual actors to actualize their conceptions of the common good/OSS product without delay. When a project's BDFL merges a "patch" that one wrote in attempt to solve an issue that "bothers" him personally, other programmers may freely act upon the same piece of code to advance their own subjective senses of a "better" product. This is the message that Elihav delivers through the emotional expression "wow!" and the assessments "cool" and "excellent" (L47:3).

Here, we also see how Ilya qua "proactive programmer" follows the normative and code rules of "la'asot cod" as a communication style. In accordance with the analysis presented in chapters 6 and 7, Ilya entered the Open Knesset project through the preferred path of online participation in the event of "terumat cod." Ilya first "forked" the source code of Open Knesset to his personal GitHub subdomain, and then "cloned" the "fork" to his desktop via the Git software application.

In so doing, Ilya did not communicate his intentions to Elihav by "asking for permission" or by "making discussions and suggestions for improvement" and therefore did not pose any risk to Elihav's scarce leisure time. Moreover, Ilya's behavior exhibits the common belief of PKW BDFLs that volunteering requires the principle of hedonic motivation by practical necessity. In Elihav's view, Ilya's behavior is an excellent example of "dookratia" because it demonstrates the unidirectional way in which participants are expected to advance their subjective conceptions of the common good. In this case, Ilya "really got into the code" as he looked for product issues that "bothered him" or "that could be bettered," turned one such issue into a programming task, and then "stepped forward and handed" the product of this task to Elihav with a laconic description from the type: I created X and this is what X does. This act of "showing code" in effect opened the social sequence of "terumat cod." By delivering his conception of a "better product" through a legible and executable computer code (rather than words), Ilya managed to "contribute to the project in seconds" and thereby gained his individual utilities – whatever these might have been.

The activity theory that accounts for such dookratic participation will need to be developed and refined in future research as it touches upon the threshold between human communication and machinic operation. Above all, the act of "showing code" reflects an attempt to change the physical world in accordance with one's preferences; it is aimed at witnesses

rather than interlocutors because it provides retroactive evidence for the merits of one's conception of a "better product" and/or for one's excellence in the virtue of proactivity.

That said, it is likely that a scholarly demand for a theory of dookratia would come not so much from philosophers of language as from political economists. Given that dookratia orients to scale the value of an unpaid workforce (in the oxymoronic sense of 'leisure time work') on the one hand, and the surprising resemblance of the organization of production it makes possible to the kind of self-regulating corporations that Marx called free association of laborers, on the other, economists and political scientists, as well as corporate executives and activists could expect to see much more of it in the contemporary and emergent world-system.

BIBLIOGRAPHY

Agha, Asif. 2011. "Meet mediatization." *Language & Communication* 31(3): 163-170

Andreoni, James. 1990. "Impure altruism and donations to public goods: a theory of warm-glow giving." *The Economic Journal* 100(401):464–77.

Axelrod, Robert. 1984/2006. *The Evolution of Cooperation*. New York: Basic Books.

Axelrod, Robert. 1981. "The Emergence of Cooperation Among Egoists." *American Political Science Review* 75:306-18.

Axelrod, Robert. 1970. *Conflict of Interest, A Theory of Divergent Goals with Applications to Politics*. Chicago: Markham.

Bakhtin, Mikhail M. 1986. *Speech Genres and Other Late Essays*. Austin: The University of Texas Press.

Bailey, Benjamin H. 2007. "Heteroglossia and Boundaries." Pp. 257-274 in *Bilingualism: A Social Approach*, edited by M. Heller. London: Palgrave Macmillan.

Barth, Frederik. 1967. "On the Study of Social Change." *American Anthropologist* 69:661–669.

Barth, Frederik. 1959. "Segmentary Opposition and the Theory of Games: A Study of Pathan Organization." *Journal of the Royal Anthropological Institute* 89:5–21.

Becker, Alton L. 1995. *Beyond Translation Essays Toward a Modern Philology*. Ann Arbor: The University of Michigan Press.

Bird-David, Nurit. 1999. "'Animism' Revisited: Personhood, Environment, and Relational Epistemology." *Current Anthropology* 40(S1)S67-S91.

Bird-David, Nurit. 1997. "Economies: A Cultural-Economic Perspective." *International Social Science Journal* 49(154):463-475.

Bird-David, N. 1993. "Tribal Metaphorization of Human-Nature Relatedness." Pp. 112-125 in *Environmentalism: the view from anthropology*, edited by K. Milton. New York: Routledge.

Bitzer, Lloyd. 1968. "The Rhetorical Situation." *Philosophy and Rhetoric* I:1-14.

Boellstorff, Tom. 2008. *Coming of Age in Second Life: An Anthropologist Explores the Virtually Human*. Princeton: Princeton University Press.

Born, Georgina. 1997. "Computer Software as a Medium: Textuality, Orality and Sociality in an Artificial Intelligence Research Culture." Pp. 139-169 in *Rethinking Visual Anthropology*, edited by M. Banks and H. Morphy. New Haven: Yale University Press.

Born, Georgina. 1995. *Rationalizing Culture IRCAM, Boulez, and the Institutionalization of the Musical Avant-Garde*. Berkeley: University of California Press.

Brown, Penelope and Stephen C. Levinson. 1978/1987. *Politeness Some universals in language usage*. Cambridge: Cambridge University Press.

Bucholtz, Mary. 1999. "'Why be Normal?:' Language and Identity Practices in a Community of Nerd Girls." *Language in Society* 28(2):203-223.

Caillé. 2013. "Anti-Utilitarianism and the Gift-Paradigm." Pp. 44-48 in *Handbook on the Economics of Philanthropy, Reciprocity and Social Enterprise*, edited by L. Bruni and S. Zamagni. Northampton: Edward Elgar Publishing, Inc.

Carbaugh, Donal. 2007a. "Cultural Discourse Analysis: Communication Practices and Intercultural Encounters." *Journal of Intercultural Communication Research* 36(3):167-182.

Carbaugh, Donal. 2007b. "Ethnography of Communication." *The Blackwell International Encyclopedia of Communication*, edited by W. Donsbach. Blackwell Publishing.

Carbaugh, Donal. 1995. "The Ethnographic Communication Theory of Philipsen and Associates." Pp. 269-297 in *Watershed Research Traditions in Human Communication Theory*, edited by D. P. Cushman and B. Kovačić. Albany: State University of New York Press.

Carbaugh, Donal. 1990. "Communication Rules in Donahue Discourse." Pp. 119-149 in *Cultural Communication and*

Intercultural Contact, edited by D. Carbaugh. Hillsdale: Lawrence Erlbaum Associates, Publishers.

Carbaugh, Donal. 1989. "Fifty Terms for Talk: A Cross-Cultural Study." *International and Intercultural Communication Annual* 13:93–120.

Carbaugh, Donal. 1988. *Talking American: Cultural Discourses on DONAHUE*. Norwood: Ablex Publishing.

Carbaugh, Donal and Tovar Cerulli. 2013. "Cultural Discourses of Dwelling: Investigating Environmental Communication as a Place-based Practice." *Environmental Communication* 7(1):4-23.

Carbaugh, Donal and Sally O. Hastings. 1992. "A role for communication theory in ethnography and cultural analysis." *Communication Theory* 2(2):156-165.

Carey, James. 1975. "A Cultural Approach to Communication." *Communication* 2:1-22.

Clark Herbert H. and Dale H. Schunk. 1980. "Polite Responses to Polite Requests." *Cognition* 8(2):111-143.

Clayman, Steven E. 2002. "Sequence and Solidarity." Pp. 229–253 in *Group Cohesion, Trust and Solidarity*, edited by S. R. Thye. Amsterdam: JAI.

Clayman, Steven E. and John Heritage. 2014. "Benefactors and Beneficiaries." Pp. 55-86 in *Requesting in Social Interaction*, edited by P. Drew and E. Couper-Kuhlen. Philadelphia: John Benjamins Publishing Company.

Coleman, Gabriella E. 2013. *Coding freedom: The Ethics and Aesthetics of Hacking*. Princeton: Princeton University Press.

Coleman, James. *Foundations of Social Theory*. Cambridge: Belknap Press of Harvard University Press.

Couper-Kuhlen, Elizabeth. 2014. "What Does Grammar Tell Us About Action?" *Pragmatics* 24(3):623-647.

Dalton, George. 1969. "Theoretical Issues in Economic Anthropology." *Current Anthropology* 10(1):63-102.

Dalton, George. 1961. "Economic Theory and Primitive Society." *American Anthropologist* 63(1):1-25.

DiMaggio Paul J. and Walter W. 1983. "The Iron Cage Revisited: Institutional Isomorphism and Collective Rationality in Organizational Fields." *American Sociological Review* 48(2):147-160

Dumazedier, Joffre. 1968. "Leisure." Pp. 248-253 in *Encyclopedia of the Social Sciences*, edited by D. Sills. New York: Macmillan and Free Press, pp., 1968.

Duncan, Brian. 1999. "Modeling charitable contributions of time and money." *Journal of Public Economics* 72(2):213–42.

Duranti, Alessandro. 1997. *Linguistic Anthropology*. Cambridge: Cambridge University Press.

Durkheim. 1893/1984. *The Division of Labour in Society*. Ney York: PALGRAVE.

Eckert, Penelope and Sally McConnell-Ginet. 1992. "Think Practically and Look Locally: Language and Gender as Community-Based Practice." *Annual Review of Anthropology* 21:461-490.

Elster, Jon. 1986. *Making Sense of Marx*. New York: Cambridge University Press

Fabien, Johannes. 1983/2002. *Time and the Other How Anthropology Makes its Objects*. New York: Columbia University Press.

Freeman, Richard B. 1997. "Working for nothing: The supply of volunteer labor." *Journal of Labor Economics* 15(1):S140–67.

Galloway, Alexander R. *Protocol: How Control Exists After Decentralization*. Cambridge: MIT Press.

Geertz, Clifford. 1984. "Culture and Social Change: The Indonesian Case." *Man* 19:511–532.

Geertz, Clifford. 1983. *Local knowledge: Further Essays in Interpretive Anthropology*. New York: Basic Books.

Geertz, Clifford. 1973. *The Interpretation of Cultures*. New York: Basic Books.

Geertz, Clifford. 1963. *Peddlers and Princes: Social Development and Economic Change in Two Indonesian Towns*. Chicago: University of Chicago Press.

Geertz, Clifford. 1957. "Ritual and Social Change: A Javanese Example." *American Anthropologist* 59:32–54.

Geertz, Clifford. 1956. *The Development of the Javanese Economy: A Sociocultural Approach*. Cambridge: MIT Center for International Studies.

Goffman, Erving. 1981. *Forms of Talk*. Philadelphia: University of Pennsylvania Press.

Goffman, Erving. 1967. *Interaction Ritual: Essays on face-to-face Behavior*. New York: Pantheon Books.

Gregory, Chris. 2009. "Whatever Happened to Economic Anthropology?" *The Australian Journal of Anthropology* 20(3):285-300.

Gudeman, Stephen and Alberto Rivera. 1990. *Conversations in Colombia: The Domestic Economy in Work and Text.* Cambridge: Cambridge University Press.

Gudeman, Stephen. 1986. *Economics as Culture: Models and Metaphors of Livelihood*. London: Routledge and Kegan Paul.

Hopper, Joseph. 1993. "The Rhetoric of Motives in Divorce." *Journal of Marriage and Family* 55(4):801-813.

Hurtado, Jimena. 2005. "The Utilitarian Foundations of the Economic Approach to Human Behavior." *SSN 1657-7191 (Edición Electrónica)*.

Hunsinger, Jeremy and Andrew Schrock. 2016. "The Democratization of Hacking and Making." *New Media & Society* 18(4):535–538.

Hustinx, Lesley, Cnaan A. Ram and Femida Handy. 2010. "Navigating theories of volunteering: A hybrid map for a complex phenomenon." *Journal for the theory of social behaviour* (40)4:410-434.

Hymes, Dell. 1972a. "Models of the Interaction of Language and Social Life." Pp. 35-71 in *Directions in Sociolinguistics: The Ethnography of Communication*, edited by J. G. Gumperz and D. Hymes. New York: Holt, Rinehart & Winston.

Hymes, Dell. 1972b. "On Communicative Competence." Pp. 53-73 in *Sociolinguistics: Selected Readings*, edited by J. B. Pride and A. Holmes. Harmondsworth: Penguin.

Irani, Lilly. 2015. "Hackathons and the Making of Entrepreneurial Citizenship." *Science, Technology, & Human Values* 40(5):799-824.

Irvine, Judith T. 1979. "Formality and Informality in Communicative Events." *American Anthropologist* 81(4):773-790.

Jones, Graham M., Beth Semel and Andrey Le. 2015. "There's no rules. It's hackathon.": Negotiating Commitment in a Context of Volatile Sociality." *Journal of Linguistic Anthropology* (25)3:322-345.

Katriel, Tamar. 2010. "Lessons Learned" (unpublished manuscript).

Katriel, Tamar. 1991. *Communal Webs Communication and Culture in Contemporary Israe*l. Albany: State University of New York Press.

Katriel, Tamar. 1986. *Talking Straight Dugri Speech in Israeli Sabra Culture*. Cambridge: Cambridge University Press.

Kelty, Christopher M. 2008. *Two Bits: The Cultural Significance of Free Software*. Duke University Press.

Keyes, Charles F. 2002. "Weber and Anthropology." *Annual Review of Anthropology* 31(1):233-255.

Lem, Stanisław. 1974. *The Cyberiad*. New York: Seabury Press.

Levy, Steven. 1984. *Hackers: Heroes of the Computer Revolution*. Garden City: Anchor Press/Doubleday.

Llewellyn, Nick. "The gift in interaction: a study of 'picking-up the bill.'" *The British Journal of Sociology* 62(4):718-738.

Locke, John. 1689/1960. "The Second Treatise of Government." Pp. 265-429 in *Locke Two Treatises of Government*, edited by P. Laslett. Cambridge: Cambridge University Press.

Malinowski, Bronisław. 1922/2002. *Argonauts of the Western Pacific: An Account of Native Enterprise and Adventure in the Archipelagoes of Melanisian New Quinea*. London: Routledge.

Marcus, George E. 2011. "Multi-Sited Ethnography: Notes and Queries. Pp. 181-196 in *Multi-Sited Ethnography Theory, Praxis and Locality in Contemporary Research*, Edited by M. A. Falzon. Burlington: Ashgate Publishing Company.

Marcus, George E. 1995. "Ethnography In/Of The World System: The Emergence Of Multi-Sited Ethnography." *Annual Review of Anthropology* 24:95-117.

Marcus, George E. and Michael M. J. Fischer. 1999. *Anthropology as Cultural Critique: An Experimental Moment in the Human Sciences*. Chicago: University of Chicago Press.

Maslow, H. Abraham. 1943. "A Theory of Human Motivation." *Psychological Review* 50:370-396.

Mauss, Marcel. 1922/2002. *The Gift: The Form and Reason for Exchange in Archaic Societies*. London: Routledge.

Merleau-Ponty, *Maurice.* 1964. *Sense and Non-Sense*. Illinois: Northwestern University Press.

Merleau-Ponty, *Maurice. Maurice Merleau-Ponty: Basic Writings*, edited by T. Baldwin. New York: Routledge.

Miller, Carolyn R. 1984. "Genre as Social Action." *Quarterly Journal of Speech* 70:151-167.

Moore, Sally F. and Barbara Myerhoff G. 1977. *Secular Ritual*. Assen: Van Gorcum.

Ober, Josiah. 2008. "The Original Meaning of 'Democracy:' Capacity to Do Things, Not Majority Rule." *Constellations* 15(1):3-9.

Olson, Mancur. 1965. *The Logic of Collective Action: Public Goods and the Theory of Groups*. Cambridge: Harvard University Press.

Ortner, Sherry B. 1984. "Theory in Anthropology Since the Sixties." *Comparative Studies in Society and History* 26(1):126-166.

Philipsen, Gerry. 2002. "Cultural Communication." Pp. 51-67 in Handbook of International and Intercultural Communication, edited by W. R. Gudykunst and B. Mody.

Philipsen, Gerry. 1992. *Speaking Culturally*. Albany: State University of New York Press.

Philipsen, Gerry. 1987. "The Prospects of Cultural Communication." Pp. 245-254 in *Communication Theory from Eastern and Western Perspectives*, edited by L. Kinkaid. New York: Academic Press.

Polanyi, Karl. 1957. "The economy as instituted process." Pp. 243-270 in *Trade and Market in the Early Empires Economies in History and Theory*, edited by K. Polanyi, C. M. Arensberg and H. W. Pearson. Chicago: The Free Press.

Polanyi, Karl. 1944/2001. *The Great Transformation The Political and Economic Origins of Our Time*. Boston: Beacon Press.

Povinelli, Elizabeth A. 1993. *Labour's Lot: The Power, History, and Culture of Aboriginal Action*. Chicago: Chicago University Press.

Rawls, Anne. 2004. *Epistemology and Practice Durkheim's The Elementary Forms of Religious Life*. New York: Cambridge University Press.

Rawls, Anne. 1992. "Can Rational Choice Be a Foundation for Social Theory?" *Theory and Society* 21(2):219-241.

Rawls, Anne. 1987. "The Interaction Order Sui Generis: Goffman's Contribution to Social Theory." *Sociological Theory* 5(2):136-149.

Raymond, Eric. 1999/2001. *The Cathedral and the Bazaar Musings on Linux and Open Source by an Accidental Revolutionary*. Cambridge: O'Reilly.

Rose-Ackerman, Susan. 1996. "Altruism, nonprofits, and economic theory." *Journal of Economic Literature* 34(2):701–728.

Rousseau, Jean-Jacques. 1775/1987. "Discourse on the Origin of Equality." Pp. 25-109 in *Basic Political Writings of Jean-Jacques Rousseau*, edited by D. A. Cress. Indianapolis: Hackett Publishing Company.

Sahlins, Marshall D. 1996. "The Sadness of Sweetness: The Native Anthropology of Western Cosmology." *Current Anthropology* 37(3):395-428.

Sahlins, Marshall D. 1976. *Culture and Practical Reason*. Chicago: University of Chicago Press.

Sahlins, Marshall D. 1972. *Stone Age Economics*. Chicago: Aldine Atherton, Inc.

Schegloff, Emanuel A. 1987. "Analyzing Single Episodes of Interaction: An Exercise in Conversation Analysis." *Social Psychology Quarterly* 50(2): 101-114.

Schneider, David M. 1976. "Notes toward a Theory of Culture." Pp. 197-220 in *Meaning in Anthropology*, edited by K. H. Basso and H. A. Selby. Albuquerque: University of New Mexico Press.

Schneider, Harold K. 1974. *Economic Man The Anthropology of Economics*. New York: Collier Macmillan Publishers.

Schudson, Michael. 2015. "The Rise of the Right to Know: Politics and the Culture of Transparency, 1945-1975." Cambridge: Harvard University Press.

Searle, John R. 1976. "A Classification of Illocutionary Acts." *Language in society* 5(1):1-23.

Shweder, Richard A. and Edmund J. Bourne. 1984. "Does the Concept of the Person Vary Cross-Culturally?" Pp. 159-199 in *Culture Theory: Essays on Mind, Self, and Emotion*, edited by R. A. Shweder and R. A. LeVine. New York: Cambridge University Press.

Smith, Adam. 1776/2003. *The Wealth of Nations*. New York: Bantam Dell.

Streeter, Thomas. 2011. *The Net Effect: Romanticism, Capitalism, and the Internet*. New York: New York University Press.

Takahashi, Nobuyuki. 2000. "The Emergence of Generalized Exchange." *American Journal of Sociology* 105(4):1105-1134.

Tkacz, Nathaniel. 2012. "From Open Source to Open Government: A Critique of Open Politics." *Theory & Politics in Organization* 12(4):386-405.

Turner, Victor. 1982. *From Ritual to Theatre: The Human Seriousness of Play*. New York: Johns Hopkins University Press

Watner, Carl, George H. Smith and Wendy McElroy. 2017. *Neither Ballots nor Bullets Essays on Voluntaryism*. Baltimore: The Voluntaryists.

Weber, Max. 1930/2002. *The Protestant Ethic and the Spirit of Capitalism*. Ney York: Routledge.

Weber, Steven. 2004. *The Success of Open Source*. Cambridge: Harvard University Press.

Winner, Langdon. 1980. "Do Artifacts Have Politics?" *Daedalus* 109(1):121-136.

Zussman, Robert. 1992. *Intensive Care Medical Ethics and the Medical Profession*. Chicago: The University of Chicago Press.