Hacking the Web 2.0

User agency and the role of hackers as computational mediators

andrea rota

Thesis submitted to the Department of Sociology of the London School of Economics and Political Science for the degree of Doctor of Philosophy.

London, January 2016
With amendments (April 2016)

I certify that the thesis I have presented for examination for the PhD degree of the London School of Economics and Political Science is solely my own work other than where I have clearly indicated that it is the work of others (in which case the extent of any work carried out jointly by me and any other person is clearly identified in it).

Abstract

This thesis studies the contested reconfigurations of *computational agency* within the domain of practices and affordances involved in the use of the Internet in everyday life (here labelled *lifeworld Internet*), through the transition of the Internet to a much deeper reliance on computation than at any previous stage. Computational agency is here considered not only in terms of capacity to act enabled (or restrained) by the computational layer but also as the recursive capacity to reconfigure the computational layer itself, therefore in turn affecting one's own and others' computational agency.

My research is based on multisited and diachronic ethnographic fieldwork: an initial (2005–2007) autoethnographic case study focused on the negotiations of computational agency within the development of a Web 2.0 application, later (2010–2011) fieldwork interviews focused on processes through which users make sense of the increasing pervasiveness of the Internet and of computation in everyday life, and a review (2010–2015) of hacker discourses focused on tracing the processes through which hackers constitute themselves as a recursive public able to inscribe counter–narratives in the development of technical form and to reproduce itself as a public of computational mediators with capacity to operate at the intersection of the technical and the social.

By grounding my enquiry in the specific context of the lifeworlds of individual end users but by following computational agency through global hacker discourses, my research explores the role of computation, computational capacity and computational mediators in the processes through which users 'hack' their everyday Internet environments for practical utility, or develop independent alternatives to centralized Internet services as part of their contestation of values inscribed in the materiality of mainstream Internet.

Contents

List of Figures

Chapter 1

Introduction

1.1 Enter the Read/Write Internet

> In retrospect, it was like I was killing my time waiting for the web to show up. [...] I sometimes feel like I was born too soon. (Jeremy Keith, in Shepherd 2013)

A small town in the suburbs of Milano, Italy. Late afternoon of the 31st of December 1995. My parents are preparing a traditional new year's eve dinner, and I have a plan on how to make best use of the long wait until midnight celebrations.

Just a few weeks before, my parents had decided to purchase a new family personal computer to replace an aging one we had been using for a few years, and I suggested that instead of disposing of the old one, we could try purchasing two network cards, connect the two computers and see what we could make of a tiny home network. My father promptly drilled a hole in the wall between the family study room and my bedroom to let a network cable run between the old computer (now in my room) and the new family computer, and brought home from a shop in Milano a small case with CD-ROMs of one of the earliest GNU/Linux distributions, Slackware 2.3, for me to experiment with setting up a 'Linux server'.

My plan for that new year's eve was to set up a web server (an early version of NCSA HTTPd), write an HTML version of the beautiful dinner menu my parents had printed out for the night, and display that on the new 'Windows for Workgrups 3.11' computer in the other room through the Netscape web browser. Sneaking in and out of the dining room inbetween the neverending succession of dinner courses, by the time we were ready to have dessert I finally had the whole setup working and was proudly showing the (likely very clunky, although no copy has survived to assess this) HTML menu to my mildly puzzled parents.

Both 'the web' and 'Linux' were very new 'stuff'—barely six and four years old, respectively. Yet there I was, a high school student with no formal training in either web or Linux, immersed in ancient Greek and Latin texts by day and eagerly trying—staying up late in the evenings—to make sense, largely on my own and by browsing some early US computer magazines that my father would purchase each month, of how information could flow between computers.

At home, we had recently started 'surfing' on the web through some early dialup ISP services, and besides the rather obvious fascination for the possibility to almost instantly search for and access information from—potentially—anywhere in the world through a connection to the Internet via our slow analog modem, I had instantly developed a strong curiosity for how this all was possible: I was rather familiar with the asynchronous store–and–forward model of data flows of early FidoNet BBSes, of which we had been occasional users at home, but what really struck a chord with me was the combination of the simplicity and elegance of the architecture of the web, and the ready availability of free software—such as my first Slackware GNU/Linux CD collection—through which an ordinary user like me could easily open the black box of software to study how the web worked in the small lab–like setting of our tiny home network.

Being able to hack on both web *content* (starting with my first web page, the HTML version of my parents' new year's eve dinner menu) and web *infrastructure* (initially, configuring the HTTPd server through which I could access my first web page through the home network) was an intoxicating experience: in hindsight, it really felt like I had been 'killing time waiting for the web to show up', and in fact developing both web content and infrastructure soon became my day job when I started my first Internet enterprise (a very 'Web 1.0' website hosting business) a couple of years later, as well as a focus of a long intellectual enquiry, through which I strived to enhance my understanding of the social, cultural and political significance of the Internet, seen as a *Read/Write Internet*: both in terms of enabling two–way information exchanges, as well as by being structurally open to being reshaped and reassembled through code and protocols.

Simple as my initial setting may have been, it nevertheless exposed very visibly the intrinsic materiality involved in reassembling Read/Write Internet even at such a small scale, encompassing countless items of everyday 'stuff' and the thick links between these: the reconfiguration of domestic spaces (drilling a hole in the wall, placing computers between other objects of daily family life and running cables to connect them through a network), vehicles of informal learning (the imported magazines and CD-ROMs of Linux distributions my father would bring home from the nearby city), hands-on experimentation (thanks to the freedoms to use, study, modify and redistribute source code granted by the licenses of the FLOSS software I was using to let the home computers exchange information over the local network and over the public Internet), feelings of joy and frustration as things would work or stubbornly refuse to do so.

As for the very modest focus and outcome of my first successful Web 'broadcast' on that New Year's eve of many years ago, one could perhaps draw an uncanny parallel between the ostensibly little practical value of my HTML dinner menu and the endless streams of latte photos that are so often criticized as a shallow use of Twitter or other web applications. This can certainly be a valid point, but—as will be discussed more in depth throughout the dissertation—it can also raise the question of whether we use the Internet for apparently pointless things, or whether our lives are actually composed also of unremarkable and very mundane but essential moments (eating breakfast, enjoying a few seconds of boredom and self–irony before moving on to focus on some challenging and perhaps important tasks), which constitute the material most easily at hand for some of our attempts to make sense of how technology may reframe our everyday lives: seen this way, my HTML dinner menu and countless moments of the life stories I came across through my fieldwork point at the importance of the material aspects of everyday life for the understanding of the processes through which we appropriate technology, which is a core focus of my dissertation.

Read/Write Internet hacking could be a part of my teenager life also thanks to my family's ability to provide these bits of simple equipment and of (at the time) physical media (GNU/Linux CD-ROMs, magazines, books): in a family context open to experimentation and independence, they served as discreet encouragement to pursue my interests by learning to dissect and reassemble software and infrastructure that until then had only been impenetrable black boxes. Although what most contributed to turning my curiosity and enthusiasm into intellectual journey and actual agency was the ability to learn from others' code and through experimentation, at the same time that learning environment vitally depended on the ability to afford a range of material tools through which to experiment: although this didn't set me apart from many other kids of the region where I grew up, who would in large part enjoy similar possibilities, in a context of rising global inequalities it undeniably put me in a small group of privileged early adopters, foretelling the crucial role of capital and the dramatic differences in spaces for agency available to individuals or small organizations on one side, and large Internet corporations on the other, throughout the recent development of mainstream Internet.

The emergence around 2005 of the Web 2.0 narrative, focused of user choice and empowerment through the ability to easily reconfigure their personal Internet experience, understandably warmed my hope that the Web could finally become a meaningful presence in individuals' lives; yet, soon after I found myself trying to come to terms with the substantial failure of my first important chance to develop a web application based on principles of user choice, technical independence and meaningful aggregation of personal content, and with the broader trend towards the simplification of the heterogeneous (and somewhat quirky but exciting in its unpredictable openness) computational complexity of early Web 2.0 experiments, within the 'walled gardens' of mainstream social network sites: these seemed to provide an increasingly convenient interface for users to deal with common everyday tasks, but without the conceptual and technical openness that those early, quirky experiments had been showing. Critically coming to terms with the failure of my first Web 2.0 project and with the trends towards centralization of control over computational lifeworld, however, constitutes the starting point of this dissertation, which can be seen as an attempt to understand the complex networks of power and negotiations of agency involved in the assembling of Internet affordances and practices in everyday life, and how—through which alliances, knowledges and practices—alternatives to mainstream Internet affordances able to effectively sustain user agency in everyday life can be attained.

1.2 Overview

The substantive enquiry of my research project starts with a small Web 2.0 project in which I was involved as lead developer between 2005 and 2007; the application I developed was a platform based on free software and designed around Web 2.0 tenets of interoperability between web applications; the aim of the application was to help students at Goldsmiths University in London with their personal development planning: while other universities were starting to offer highly prescriptive exercise–based environments in order to comply with funding bodies' requirements to put systems in place for students' personal development planning, my design sought to enable students to manage their development notes, plans, portfolios and other relevant content through any web apps they would be already using for these purposes or would wish to try using, while the web application I developed would offer flexible ways to bring all the relevant information together through personal 'mashups'.

As a curious user of early Web 2.0 applications myself, I had been signing up to every new Web 2.0 app I would come across, trying to imagine if and how each of these could somehow fit within my daily tasks, allowing me to organize some aspect of my everyday life efficiently or to provide some useful features such as helping me to keep track of my growing collection of photos or to discover new music through recommendations based on similarities with other users' playlists. When I had the opportunity to imagine how to help other students to make sense of their disparate everyday activities, plans and achievements through a digital platform, using Web 2.0 strategies such as information mashups seemed like a perfect match for the specific scenario: my application would only need to let students aggregate content in ways they deemed meaningful, while they could choose freely, amongst the many Web 2.0 apps available at the time, which ones to use to manage their own content—different students would have different priorities and different tastes for specific user experiences and information aesthetics, and therefore letting them feel comfortable with their choices of web apps would have helped to make my content–aggregation app relevant.

However, once development started, a multitude of conflicting agendas, technical difficulties and—crucially—a substantial lack of interest amongst students for actually experimenting with using web apps at all, slowly led the project to ultimately failing to deliver a meaningful Web 2.0 experience to students; in its place, my office pragmatically introduced a set of standardised forms that students were invited to download, print and fill in. What I felt at the time to be a disappointing personal failure, however, once critically dissected revealed how—within and beyond the local context—the domain of Internet affordances related to everyday practices was already being shaped by wider systems of power relations (social, epistemic, institutional, technical) that enabled some Web 2.0 configurations to attain mainstream relevance, while relegating others to ultimate lack of relevance and usefulness.

Moreover, as the Internet at large quickly evolved in the following years, some of the traits of the Web 2.0 narrative related to personal user content ended up being redefined within large–scale commercial and proprietary social network sites: most notably—in the context of Goldsmiths—MySpace at first, then Facebook, with some of the original ideas about interconnection of small, specialised web components being reshaped along the way as parts of vast corporate–controlled Internet 'walled gardens'. Although the Web 2.0 narrative kept influencing the tech scene, the more generic and empowering flair of user–centered Internet was lost along the way: normative configurations prevailed, both in the case of my development project and across the Internet. In order to explore the implications on user agency of the ongoing consolidation of everyday life within mainstream Internet, I returned to the same fieldwork site in 2010–2011 to interview students about their use of the Internet in everyday life. Most of them were now very active Internet users and although the majority was using web apps and services that didn't provide much opportunity for reconfiguration, their accounts highlighted that they were generally satisfied with their ability to meaningfully appropriate these apps and services in their everyday life.

When taking a step back from the letter of these users' accounts, however, one could not fail to notice that despite the satisfaction reported, most users were spending most of their daily Internet time using websites over which they exerted very little control besides choosing whether to use them or not and toggling a few cosmetic settings: all their data and content, and the software code used to transfer, store, combine and present them, was typically controlled by some large corporation or Internet startup, often without even a possibility for users to export their own personal content for reuse elsewhere.

As a developer that had been involved in free software projects for over a decade by then, and as an early user of Web 2.0 applications enthusiastically imagining ways in which free software apps and data and content streaming between them could empower both 'power users' and common users to meaningfully make use of the Internet in everyday life, the ongoing consolidation of control within the information silos of private corporations seemed to me like a missed opportunity. I could understand the allure of Facebook's addictive voyeurism and even appreciate the opportunity to easily keep in touch with friends and family, and I could see how the user experiences of the most popular social network sites and other mainstream web applications were in most cases vastly more aesthetically pleasant and understandable than those of early free software counterparts that lacked dedicated design resources (and often even proper plain software coding resources), but the thought of putting one's life (or at least, the data associated with one's life) in the hands of ultimately unaccountable private organizations kept making me uncomfortable.

My research motivations can therefore be seen as laying at the nexus of a political interest in the unfulfilled potentials of the Internet, further informed by the motivations and justifications present in the accounts of my fieldwork participants, and the historical trajectory of the mainstream Internet which privileged normative configurations: the Web 2.0 'didn't happen' alongside its more progressive potentials, yet at a micro scale—that of individual users interacting with Internet technologies within the lifeworld horizon of their everyday life—the specific reconfigurations of affordances observed highlighted ingenious ways to circumvent limits imposed by the dominant configurations of the of Internet that have been taking shape instead of explicitly open configurations.

Following a material culture approach, my interest lies in understanding how the Internet—alongside other everyday 'stuff'—makes users as much as users make the Internet, at several distinct layers, ranging from the most immediate and personal reconfigurations of technical form in everyday life, to contributing content through some Web 2.0 apps, to appropriating specific Internet apps and attempting to reconfigure Internet infrastructure with the aim of negating the limits and exogenous constraints that some users perceive in the shape of mainstream commercial Internet.

A focus on the material culture of the Internet also extends to the methodological choice of approaching this enquiry by following, on one hand, individual users' struggles with technology, their ways of choosing, connecting and composing technological affordances and their justifications and explanations for their choices and practices, while on the other hand including in the analysis the issues of power and agency stemming from specific historical, situated configurations of the complex networks of human and non–human actants that shape the infrastructure of the Internet; these configurations constitute both the technological background within which users' practices can take shape, as well as the boundaries that delimit the agency of users and the domain of possibilities available to them.

One crucial factor that emerged through my research is the fundamental reliance of recent Internet on the ability to *execute* code: although the focus on user–generated content of Web 2.0 narratives and the dominant representations of the Internet as medium through which information and data is exchanged contribute to hiding this factor within the black–boxing of Internet infrastructure, the analysis developed here highlights how several of the power imbalances becoming manifest on the Internet may be traced back to uneven availability of knowledge or capital, but are often ultimately mediated by the material ability to execute code. This could be as apparently simple as the algorithms that route data packets across the Internet infrastructure, consisting of short functions that take decisions in some cases millions of times per second on a single busy Internet router, up to complex and large code bases running on cloud infrastructure, on which web applications and APIs rely; almost every user interaction on the Internet relies on a combination of data transmission and code execution at multiple layers (from infrastructure physically remote from end users, up to the web browser through which a user accesses a web application): somewhat presciently, O'Reilly's (2005b) seminal article on Web 2.0 highlighted emerging shifts in software engineering and web operations aimed at improving efficiency of code execution from a corporate point of view, and as I discuss in Chapter 4, since around 2010 mainstream Internet has in fact been relying much more decisively on the ability to execute code, across all layers of infrastructure and across an increasingly wide array of types of devices.

It is useful to briefly introduce at this stage the key concept of 'computational turn of the Internet' that will be developed in detail in Chapter 4, but on which earlier discussions rely. The idea of a 'computational turn' may seem counter-intuitive in this context, as the Internet is inherently reliant on digital transmission of information and on execution of code through digital devices, whether at the layer of infrastructure or on user devices. Referring to a 'computational turn' may indeed seem to be intuitively more appropriate when focusing on disciplines that have existed long before (digital) computation, such as the humanities, on which Berry's (2011) analysis is focused. However, as I will elucidate in Chapter 4, a profound shift can also be seen as having happened within the timeframe of my research (2005–2015), making the fabric of the Internet in 2015 vastly different from that of the Internet in 2005, and proportionally much more reliant on computation at this later stage. Web 2.0 apps as developed around 2005 were certainly possible only thanks to web servers running the apps' code, databases and other digital infrastructure; the basic architecture of this infrastructure, nevertheless, was not dissimilar from that of earlier web (or non-web: for example, corporate enterprise resource planning systems) applications. Gradually, however, over the past decade the scale of Internet software has become unsustainable without the support of computational tasks at different levels: producing, testing and deploying software code (e.g. by using software that checks the formal correctness of any new code and that helps large groups of developers to effectively collaborate on complex code bases); setting up Internet servers on which code is run (e.g. by using 'cloud services', which in turn rely on software code to abstract physical hardware into 'on–demand' computational capacity); coordinating large pools of servers (e.g. by using software that ensures that extra computational capacity is made available almost instantly whenever the load of the existing infrastructure is unable to cope with spikes in Internet traffic); running code efficiently within users' browsers (which since around 2009 have become very complex computational platforms able to run efficiently code that only a few years before could reasonably only be run on specialized servers); managing code execution (apps) on mobile devices, which have as well become very powerful computational platforms that users can carry around through their daily life, etc. On one hand, large software ecosystems—such as the Facebook platform, the Google Android platform, the Apple iPhone/iPad platform—could not exist without the computational complexity that automates most of the management of the infrastructure that supports them; on the other hand, and at the same time, these principles of computational management of infrastructure and code have made computational complexity once domain of large organizations accessible to small startups and to independent free software developers, who can rely on it to develop and manage services and applications focusing on their ultimate goals (solving specific real–world issues, providing alternatives to mainstream services, etc.) without having to invest considerable resources in the management of the underlying infrastructure. 'Computational turn', therefore, is articulated here as a shift to a qualitatively and quantitatively different reliance on computation at every level of the structure and infrastructure of the Internet, rather than a transition from an analog to a digital fabric as in the case of the humanities, to reconnect to Berry's analysis referenced above: the aim of the detailed discussion of the computational turn of the Internet developed in Chapter 4 is to identify the broader cultural, epistemic and political significance of this transition in relation to the research focus of my work.

The centrality of computation and computational capacity to contemporary Internet is not only a pivotal moment in the recent history of the Internet itself but also a key concern of my dissertation: the argument that I develop throughout the last two empirical chapters is that the computational turn of the Internet and the concurrent (but only partly interrelated) constitution of a new visible 'recursive public' (Kelty 2008) of hackers are the two most important factors for the progressive opening of spaces where alternatives to the hegemonic architecture, infrastructure and affordances of mainstream Internet can be developed. On one hand, as just discussed, the easy and relatively affordable access to computational capacity that has allowed large Internet and advertising corporations to establish web services used by millions or billions of users is often accessible, on a much smaller scale, to independent developers and projects who need to exploit efficient computational capacity to sustain alternatives to mainstream Internet affordances; on the other, hackers constitute themselves as computational mediators, negotiating the shaping of affordances, reproducing hacker knowledges, cultures and values (I use here the plural to visibly signpost the multiplicity of often conflicting ideological stances and agendas—a discussion that I develop in Chapter 7—beyond a common interest in constituting themselves as a recursive public) through discursive processes often supported by computation itself, and by doing so contribute to shaping their own computational agency as well as that of common users.

As the last step of my enquiry, I review actual alternative rationalizations developed by hackers intent in contesting the proprietarization and centralization of computational agency within lifeworld Internet—the domain of Internet practices and affordances that relate to the everyday life of users. Through this final analysis I reconnect my enquiry back to its origins in the exploration of what now appears a very different kind of openness—the one promised by the Web 2.0 narratives, and effectively often determined within proprietary boundaries—by articulating how alternative rationalizations try to reconfigure spaces for user agency that can avoid the uncomfortable compromises that were involved in my initial Web 2.0 development project and that emerged from the accounts of the students I interviewed.

1.3 Research focus and research question

As outlined in the previous section, the domain on which my enquiry is focused is that of Internet practices related to the everyday life of individuals. This focus stems primarily from my interest in this specific domain, from the design of the Web 2.0 application around which the first phase of my fieldwork was developed, and from the accounts of students I interviewed in the second phase of my fieldwork. Throughout the rest of the dissertation I refer to this domain as *lifeworld Internet*: the intent is not to articulate an original concept but to identify this domain through a manageable label wherever I need to clearly signpost that the practices and affordances I am referring to are the ones related to the basic, mundane, often unglamorous 'stuff' that users do in everyday life on the Internet, rather than what most readers would be familiar with or assume when dealing with Web 2.0 and Read/Write Internet practices, such as user–generated content, collaborative platforms, wikis, micropolitical participation, radical politics and civil unrest (Castells 2012; Dahlgren 2013; Fuchs 2014b; Gerbaudo 2012), etc. Conceptually, the closest existing category in scholarly literature is Bakardjieva and Gaden's (2012) 'Web 2.0 technologies of the self', which in turn references Foucault's analysis (1988) of 'technologies of the self'. However, Bakardjieva and Gaden's analytical focus on Web 2.0 technologies, albeit related to my own, would be a confusing reference when used in the many contexts where I am not dealing with Web 2.0.

Relatedly, I also strive to analytically distinguish proper mentions of Web 2.0 practices and affordances from the analysis of practices and affordances that, while in some cases superficially similar, would need to be kept distinct from the ideological references implicit in Web 2.0 narratives (these are deconstructed in detail in the next chapter). Moreover, whereas a large part of O'Reilly's (2005b) articulation of Web 2.0 revolves around the role of Internet infrastructure (which is a central concern of my approach), his discussion nevertheless relegates agency related to infrastructure exclusively to professional actants (developers, software engineers and other web professionals, including the marketing, legal and financial roles part of the corporate production of Web 2.0 applications), whereas end users are mainly seen as only able to participate in the Web 2.0 narrative through their production of, and interaction with, content, and through essentially predetermined options for a reconfiguration of their computational environments. The resulting sharp demarcation of the domains over which professional and non–professional actants are seen as being able to exert agency is negated both by my fieldwork findings and by my analysis of the empirical materials emerging from hacker discourses, making the Web 2.0 label even more problematic to use as a proxy for what is essentially a *Read/Write Internet*: through this terminology I refer to user–facing and infrastructural technologies, practices and narratives involved in the use and in the reassembling of the Internet as a two–way medium.

Central to this definition is therefore the tenet that the protocols, code and physical infrastructure that sustain the Internet are potentially contestable and therefore susceptible to being redefined (often in non–rival ways: alternative implementations can coexist) like any other digital 'content' that is handled by users in read/write practices over the Internet; this idea includes the view that computer code is substantially data, which is executed rather than being otherwise manipulated (Berry 2014; Floridi 2014), but extends beyond software itself to include in the analysis of contested practices the network layouts (e.g. the use of VPNs and mesh networks), storage of personal information (e.g. the IndieWeb), etc.

Whereas user practices and hacker interventions constitute the primary empirical materials on which my research is focused, within the domain of lifeworld Internet as articulated above, the actual core research focus is on *computational agency* as it pertains to both users and hackers; computational agency is here analyzed not only in terms of capacity to act enabled (or restrained) by the computational layer, whether pertaining to human or machinic agents, but also as the recursive capacity to reconfigure the computational layer itself, therefore in turn affecting one's own and others' computational agency.

The main research question that my analysis strives to address is *how computational agency is (re)configured within the domain of lifeworld Internet, throughout the transition of the Internet towards a much higher reliance on computation than at any other previous time*.

There are several aspects to this question, which are taken up throughout the present dissertation and which, together, inform the specific approach to the study of computational agency in lifeworld Internet. Firstly, the domain of lifeworld itself is essentially fluid and porous, carrying different meanings to different individuals and often to the same individual in different moments and contexts: whereas my main intention is to maintain a distinction between the participatory practices on which part of Internet scholarship focuses and the practices that pertain to the private sphere, some of my research participants clearly remarked that participation to micropolitical activism was indeed part of their everyday life and therefore—inasmuch as they were using Internet affordances to participate in it—also part of their lifeworld Internet. Accordingly, what users expect to be able to do through lifeworld Internet affordances—and therefore the kinds of computational agency they seek to attain—constitute a broad spectrum of desired configurations, that can be addressed through different strategies. Secondly, although I have been keeping a lexical distinction between users and hackers so far, both through my local fieldwork and through my analysis of hacker discourses I observed that there is often continuity between the two analytically distinct domains, and this is in fact a recursive effect of the specific flairs of hacker public and hacking practices that have been taking shape through the computational turn of the Internet: whereas Web 2.0–style user agency was typically realized through knobs, buttons, drag–and–drop gestures and other visible and manipulable affordances on screen, caring for alternative rationalizations essentially requires an understanding of, and an interest in being involved with, the intersection of the social and the technical: alternative rationalizations are not only practical implementations of code outside of the boundaries of what is 'normal' in mainstream Internet, but also, and more importantly, political statements about the ability, the intention and the power to reshape technical form according to counter–narratives to hegemonic discourses.

Accordingly, the methods required for the exploration of this research question are necessarily multidisciplinary and require access to the lifeworlds of different groups of actants, while keeping a constant reference to the common grounding in the concept of computational agency: as will be articulated in the following outline, the second and third empirical chapters are based on traditional ethnographic methods within a local context, whilst the two final ones involved a broad review of hacker discourses through a multiplicity of sources and across disciplinary boundaries: whereas most of the relevant materials were personal reflections of hackers on their own role, on issues of power and control, on broader societal issues, technical discourse was often tightly intertwined, and unavoidably I also had to recur to highly technical literature that could clarify the intricate details of technical form in order to assess how alternative rationalizations were sustaining, in practice, hackers' stated goals and principles. Furthermore, exploring this research question entails articulating, as a whole, the complex set of interactions between software, devices, their users and their developers (and the various overlaps of these groups) through social practices, cultural representations and technical materiality, following the ways in which lifeworld Internet is shaped by these and in turn shapes them: 'Another way to state this is that software as code, behavior, and structure both *expresses* relationships and *organizes* those relationships' (Ratto 2005, p212).

In an ethnographic sense, the question itself, even before any possible responses, has been driven by the fieldwork participants and their accounts: the initial assumption that I could study behaviours, practices, motivations on and around the Internet presumed that the Internet itself could be somehow defined and circumscribed as a technological environment ('whatever is carried on the TCP/IP protocol suite') and as a social and cultural environment ('the practices involving access to the Internet'). However, through my conversations with fieldwork participants, it became increasingly clear that in following the actors I was not only following the participants but also (in an actor–network sense) a multiplicity of 'things', some of which are the multiple representations of 'the Internet' which are ultimately most relevant to each individual's daily experience, alongside the overall framework of norms and discourses surrounding 'the Internet' as a shorthand for its actual richness and intricacy. What I actually observed is that most of the times, 'things' that are clearly not happening while connected to the Internet, such as individual aspirations, frame–changing life events and traumas, expectations about how to manage personal time and resources, are fascinatingly influencing the ways in which users experience, practice and shape their lifeworld Internet, hinting at the fact that not only—as common sense could suggest—the Internet is colonizing people's lives, but also people's lives are colonizing the Internet, therefore making the issue of mapping spaces for computational agency unencumbered by the centralized control of private corporations all the more urgent and relevant.

1.4 Thesis outline

In **chapter 2** I review the scholarly literature relevant to the themes analysed in my thesis and introduce key concepts and terminology used in the following chapters. This chapter is composed of four sections. In the *first section* I trace the evolution of scholarly studies of the 'pre-Read/Write' Internet, highlighting aspects of earlier representations (foremost, a distinct split between *online* and *offline*, as well as variously declined expectations of freedoms related to the extension of interpersonal communication through the Internet) whose legacy was still firmly present in the accounts of my fieldwork participants. In the *second section* I focus on key concerns of the academic debates around technologies and practices close to my core research focus on computational agency on the Internet: how the transition from an early (mostly) read–only Internet to a platform enabling high levels of interactivity has been interpreted in its various aspects (social, cultural, technical, economic) in recent literature. In the *third section* of this chapter I discuss how my focus on the domain of lifeworld Internet can be reconduced to the articulation of the concept of lifeworld in phenomenological theory, and how the centrality of computation and computational agency in my analysis both complicates and questions the role of communicative action in Habermas' theory. In the *final section* I discuss the theoretical framework employed throughout the present study, underlying both the approach to the fieldwork and the choice of methods, as well as the analysis of the discursive materials gathered through my fieldwork.

In **chapter 3** I discuss the research methods employed in my research and the reasons for their choice. This chapter is composed of three sections. The *first section* describes the site and contexts of the initial autoethnographic case study and of the successive ethnography of user accounts. The *second section* looks at the methodological considerations involved in the process of recruiting and interviewing students, and examines how the different levels of technical expertise encountered were taken into account and contrasted to the ideal–type of the ordinary user. The *third section* looks at the different dimensions through which I attempted to make sense of fieldwork findings: here I analyze how prior knowledge and my role as both researcher and technical expert was taken into account, and to what extent the different parts of my fieldwork were informed by autoethnographic methods. I also describe which materials informed the analysis of hacker discourses and practices developed in chapters 7 and 8.

The core empirical material of the dissertation is developed throughout five chapters (Chapter 4 to Chapter 8).

The first empirical chapter (**Chapter 4**) examines the materiality of mainstream Internet, with a specific focus on the domain of *lifeworld Internet*: the sphere of practices and technical affordances that involve the use of the Internet throughout the everyday lives of common users. I examine connections to the early Web 2.0 narrative and its successive developments, as well as to the historical process through which a client/server architecture—rather than a peer–to–peer one—prevailed across mainstream Internet, sustaining specific kinds of practices and of agency. Secondly, the analytical framework of *computational turn of the Internet* is introduced, arguing that across the timespan of my empirical research a decisive shift happened, making the Internet in 2015 much more vitally reliant on computation than it was in 2005 (or ever before). The central argument is that the transition from Web 2.0 promises of openness and choice to the recent mainstream configurations as well as the potential for development of alternative rationalizations both revolve around the role of computation as the *site of mediation between the social and the technical*: through it, values and norms of relevant social groups are negotiated and inscribed in technical form, configuring spaces for computational agency that are historically and materially determined and contested. This chapter is composed of four sections. In the *first section* I trace how the architecture of lifeworld Internet is sustained by Read/Write affordances, in turn rooted in the early Web 2.0 narrative of 'network as a platform'. In the *second section* I build upon this articulation of human and non–human actants in order to analyze how the telos of progress that permeates Web 2.0 discourses, visibly promoted through the rhetorical artifice of version numbers (2.0, etc.) describing successive evolutionary stages of technological form, contributed to formulating a one–dimensional narrative that privileges the idea of linear evolution and by doing so colonized mainstream lifeworld Internet, while relegating to subaltern position alternative models of computational agency. In the *third section* I analyze the tensions between two different topologies of infrastructure—a client/server architecture and a peer–to–peer architecture—as well as the historical processes through which the client/server architecture became the largely dominant configuration, in turn determining a range of ways in which individuals could appropriate the Internet as a two–way medium within the domain of everyday life. In the *fourth section* I introduce the key traits of the computational turn of the Internet as can be observed in recent (post-2010) discourses and practices of web and Internet software engineers, and I develop the analytical framework of computation as site of mediation between the social and the technical, which is used throughout the following chapters.

The second empirical chapter (**Chapter 5**) examines the case study of my work on a Web 2.0 application aimed at helping university students to critically reflect on their personal development within a digital information environment. The relevance of this case study to the research topic of the dissertation is due to the fact that it ultimately failed: by framing it as an example of sociology of translation (Callon 1984), the failure to engage human and technical actors across the four key moments of problematisation, interessement, enrolment and mobilisation helps to disclose the diverging interests and forces that are here related to the wider systems of power that enabled some Web 2.0 configurations to gain mainstream relevance, while relegating alternatives to niche status. What at the time felt like a frustrating and shameful personal failure is here revisited to position it within the analytical framework of decisive computationalization of the architecture of lifeworld Internet. The chapter is divided into five sections. The *first section* outlines the original design of the application, highlighting how the emerging narrative of Web 2.0 information mashups was conjugated in practice to imagine an environment focused primarily on the lifeworld of individual students rather than on the institutional framing of personal development planning. The *second section* examines the tensions between my attempts to let students decide how to use the guidance for self reflection provided by the web application, on one side, and the institutional requirements related to dissemination of knowledge and to performance metrics, on the other side. The *third section* looks at the social and technical challenges that emerged through the development of the web application; the aim is to critique the complicated relationship between the delicate and unique peculiarities of user lifeworlds, and the technical form that was already being architecturally shaped to accommodate the walled gardens of corporate SNSes. The *fourth section* looks at the materiality of technical constraints within which the web application was developed: the building blocks (tools and software libraries) and the software engineering best practices available at the time carried the inscription of specific and limited ways of organizing knowledge and of enabling social interactions mediated through software; the aim is to look at how these material inscriptions affected the development of Web 2.0 applications that wanted to deviate from nascent yet already hegemonic rationalizations. In the *fifth section* I review the final shape of the project, analyzing the negotiations and the compromises that led to a much less ambitious technical form as well as to limited educational and social relevance of the project.

The third empirical chapter (**Chapter 6**) explores, through the accounts of the students I interviewed for my fieldwork, the material extent and shape of what is constituted as lifeworld Internet throughout the actual practices of individuals, and how individuals' computational agency is configured within this domain of lifeworld Internet. The timing of the fieldwork on which this chapter is based (2010-2011) coincides with the pivotal moment of the computational turn of the Internet discussed in Chapter 4: although this was not yet visible at the time, user attitudes and practices discussed by my research participants were already being reshaped by this transition.

This chapter is composed of three sections, each of which looks at a distinct trait of computational agency within the domain of lifeworld Internet. In the *first section* I review my assessment of how the original Web 2.0 promises related to the ability to easily create and share content and to configure personalized Internet environments thanks to the interoperability of web applications—both assumptions on which my Web 2.0 project (Chapter 5) relied—had actually translated into an increased capacity for common users to integrate disparate aspects of their Internet–connected lives. In the *second section* I look at computational agency through the tension between norms and actual practices as discussed by students in relation to their use of the Internet. In the *third section* I focus on how computational agency was being reconfigured by and around users at the time of my fieldwork; on one hand, I analyse how users were making sense of the growing array of applications and devices available to them and of their ability to configure a computational environment meaningful to them; secondly, I look at deeper relationships with code and computation through the accounts of some students who disclosed some significant degree of technical expertise and discussed at length the benefits they perceived from the ability to deeply re–assemble their Internet experience thanks to the possibility to study, 'hack' and appropriate free/open source software, beyond the reconfigurations operated by other students who were simply choosing and combining, without complex modifications, available affordances.

In the two final empirical chapters (Chapter 7 and Chapter 8) I shift my focus to actual counter–narratives and alternative rationalizations of lifeworld Internet, relating back to the affordances and practices relevant to the user experiences observed throughout my local fieldwork, in order to examine how these alternative rationalizations may enable different configurations of computational agency that overcome some of the external constraints analyzed in the two previous chapters, allowing common users to exert control over their lifeworld Internet beyond the range of possibilities available through Internet affordances and related narratives as determined by a largely capital–driven mainstream development.

In **Chapter** 7 I analyse the constitution of hackers as an increasingly visible 'recursive public' (Kelty 2008), acting as computational mediators at the interface between the technical and the social, establishing and expanding spaces where user and developer discourses are able to effectively shape the development of technical form, and developing actual affordances aimed at addressing the power imbalances inscribed in mainstream Internet. The chapter is composed of three sections. In the *first section* I look at strategies through which hackers recursively inscribe in technical infrastructure the ability to take into account user requirements rather than forcing these to be reframed within the limited options available under mainstream technical configurations. The aim is to identify how complexity of real world use cases is reflected in computational complexity through the public mediation of hackers, as opposed to earlier top–down approaches (as exemplified by the case study of Chapter 5). In the *second section* I focus on hacker learning and on reproduction of hacker knowledge: my argument is that through wide accessibility of learning resources hacker knowledge is being accumulated, reviewed and reproduced at a global scale, resulting in improved opportunities for hackers to learn from quality resources and to attain capacity to develop software meaningful to users. In the *third section* I look at how hackers are progressively constituting a new global recursive public intent in understanding the implications of its own capacity to operate at the intersection of the technical and the social through the role of computation.

In **Chapter 8** I look at actual alternative rationalizations through which hackers shape affordances that enable users to exert computational agency within lifeworld Internet outside of the limited options provided by mainstream Internet affordances. Among a wide array of alternative rationalizations, I focus on two domains that are specifically related to the limitations and constraints that emerged through the analysis of my local fieldwork, as experienced by myself as a Web 2.0 developer and through the accounts of my research participants: alternative rationalizations are here analyzed not just as alternative implementations of mainstream affordances but as attempts to redress power imbalances inscribed in mainstream centralized configurations, through the role of the recursive public of hackers (discussed in the previous chapter) as cultural translators and computational mediators. This chapter is composed of three sections. In the *first section* I analyze the domain of practices related to self–hosting (and controlling/owning) one's own content on the Internet—rather than having to rely on walled gardens and information silos controlled by corporate entities and often provided through proprietary software. Being able to meaningfully control one's own content was the core tenet of the design of my Web 2.0 application whose development struggles have been discussed in Chapter 5, and similarly the accounts of my research participants highlighted their practices related to meaningfully appropriating Internet affordances while dealing with personal content. The *second section* deals with the domain of tensions between the centralization trends of mainstream Internet discussed in Chapter 4 and the 'redecentralization' efforts of hackers concerned with the imbalances of power and control over the lowest, infrastructural fabric of lifeworld Internet that stem from centralization of computation and infrastructure. The *third section* concludes my overview of alternative rationalizations by examining the struggles involved in operating outside of the boundaries of mainstream narratives and technologies: how these alternatives are represented as relevant to common, non technically–minded, end users and how issues (such as management of identity and trust) that are efficiently managed through centralized configurations are addressed within decentralized alternatives.

In my final conclusions (**Chapter 9**) I summarise the core findings, arguments and contributions of the previous chapters, reconnecting them to the core research question of my dissertation. Finally, I propose possible future directions of research that could further explore the articulation of computational agency in lifeworld Internet.

1.5 Contributions to previous research

Through my research I contribute to the existing academic literature in several ways. Firstly, I present *novel substantive materials* about use, development and re–assembling of lifeworld Internet, through my fieldwork interviews (Chapter 6) and through the critical deconstruction of my own development work on a Web 2.0 application whose design was meant to empower students to reflectively bring together diverse streams of personal content (Chapter 5). Both case studies focus on user agency in the specific context of my fieldwork: respectively, individual students and a small–scale software engineering process; while doing so, however, I actively engage with the wider context of computational structures involved in the assembling of mainstream Internet, in order to situate the fieldwork findings within the analytical framing of computational turn of the Internet articulated in Chapter 4. Most of the substantive materials underpinning the analyses of Chapters 7 and 8 are in large part publicly available (academic and technical literature, as well as hacker discourses articulated through engineering blogs, personal blogs, discussion forums, magazines and conference presentations), with the exception of materials gathered through my participation in a few hacker events. As most of the hacker discourses examined here are relatively new (2010–2015), however, only very recent literature is starting to make use of them: my dissertation is a further contribution to this recent set of scholarship.

Whereas not an innovation per se, the domain of empirical focus of my research ('lifeworld Internet') is however less explored than other domains of human activity on the Internet: the substantive focus is on Internet practices related to the mundane and intimate everyday life of individuals, within the spheres of well–being, sociality, education and culture, whereas practices related to work, micropolitics/subactivism (Bakardjieva 2012), Internet–mediated collaborative endeavours, as well as activity in virtual worlds (Boellstorff 2008) and videogame play, however widespread and subject of scholarly interest, are only incidentally mentioned where reported as relevant by individual research participants.

The main *theoretical contributions* I strive to make with my dissertation are the key conceptual framings that sustain the interpretive stance of my work. Firstly, the articulation of *computational turn of the Internet* as a fundamental transition towards a much deeper reliance of the technical form of the Internet on computation than at any previous stage, which in turn sustains both mainstream practices of centralization of control as well as the ability for independent developers to leverage computational complexity to develop alternative rationalizations. Secondly, the analytical framing of *computational agency*, which is here considered not only in terms of capacity to act enabled (or restrained) by the computational layer, whether pertaining to human or machinic agents, but also as the recursive capacity to reconfigure the computational layer itself, therefore in turn affecting one's own and others' computational agency.

The *methodological contribution* of the present work, similarly to Kelty's (2008) consists in 'an example of how to study *distributed phenomena* ethnographically' (2008, pp19–22), although the actual focus is rather different from Kelty's study of the cultural significance of free software: he notes that

> Free Software and the Internet are objects that do not have a single geographic site at which they can be studied. Hence, this work is multisited in the simple sense of having multiple sites at which these objects were investigated [...]. It was conducted among particular people, projects and companies and at conferences and online gatherings too numerous to list, but it has not been a study of a single Free Software project distributed around the globe. (Kelty 2008, p19)

Through my work I analyse the distributed phenomena involved in the contested reassembling of lifeworld Internet, although, given the focus and scope of my research, my methodological interest lies less in the topology and diversity of the multiple sites and in the strategies to fruitfully explore the spatially fragmented network of sites: rather, it is focused on phenomena whose distributed character in terms of classes of actants (capitalist entrepreneurship, developers, end users) is more relevant than their being distributed geographically. My strategy to address the distributed nature of lifeworld Internet methodologically is therefore trying to address the issues inherent in the analysis of heterogeneous publics whose ability to influence the shape of Internet affordances ranges widely from hegemonic to subaltern positions: the aim is to cross (Marcus 1995, p96) boundaries between lifeworld and system (here, respectively, the fieldwork participants' making sense of their use of lifeworld Internet in everyday life, and the global tensions and contestations related to the inscription of different agendas and networks of power in the technical form of Internet infrastructure), acknowledging that this is itself one of the potentialities inherent in the contested practices of Internet development and a necessary way to approach its study as an environment of distributed phenomena:

> [This ...] mode of ethnographic research self–consciously embedded in a world–system, [...] moves out from the single sites and local situations of conventional ethnographic research designs to examine the circulation of cultural meanings, objects and identities in diffuse time–space. [...] Just as this mode investigates and ethnographically constructs the lifeworlds of variously situated subjects, it also ethnographically constructs aspects of the system itself through the associations and connections it suggests among sites. (Marcus 1995, p96)

Research that interfaces with critical software studies and material culture of the Internet is particularly relevant and useful to the understanding of the relationship between technical and social that is constitutive of daily life in most of the world today. Whereas through the field of Internet studies important contributions have been made to the understandings of how users incorporate the Internet in their everyday lives and how people act online, only more recently through the growing field of software studies a closer understanding is being sought of the materiality of the contested development of the infrastructure that sustains users' practices on the Internet. If a parallel could be suggested with the rich field of urban studies, we have been studying in depth the places where people gather, what they do there and how urban form and technologies (Greenfield 2013) affect the personal sphere and the social, while neglecting the ways in which urban form is negotiated, built and reshaped: my thesis aims to contribute to the understanding of these practices of negotiating, building and reassembling lifeworld Internet as a part of the fabric of contemporary life.

A final element of distinction of my research interfaces with all the three domains of empirical materials, conceptual framings and methods: my own role as researcher and developer involved for two decades in free software discourses has hopefully allowed me to provide a much closer reading of technical materiality than would have been possible if I were a researcher not exposed daily to technical discourses in a professional capacity; whereas obviously social researchers with any kind of technical or non technical background can meaningfully explore technical materiality, I hope that my role of ‘native’ has enabled me to build an original interpretive framework closely grounded in the details of computation.

Chapter 2

Conceptual framework

Throughout this chapter my aim is to outline the scholarly literature relevant to my dissertation's core research question, as well as to introduce key concepts and terminology used in the following chapters. I propose the label of Read/Write Internet as reference framework for affordances and practices relevant to my research and analyzed in previous literature, often using the label of Web 2.0; the reason for introducing a new label is that my focus on Read/Write affordances and practices coincides only in part with the narrative of Web 2.0, which is also ideologically charged (as discussed in the second section of the chapter) and therefore unsuitable for the analysis of user and developer agency that lie outside of the historically determined boundaries of the Web 2.0 narratives of user choice and ability to reconfigure users' web experiences. Moreover, Read/Write Internet as articulated here keeps an analytical distinction between two domains of user agency: firstly, the one related to affordances and practices involved in the use of the Internet as a two-way medium; secondly, the one related to users' ability to reshape the technical form of Internet affordances according to their individual needs and tastes, according to their capacity to act on technical form, and within the possibilities for reconfiguration that pertain to the affordances themselves.

My aim throughout this chapter is to frame the central issue of my dissertation — computational agency of users and developers within the domain of lifeworld Internet and across the *computational turn* of the Internet — within the existing scholarly literature that looks at spaces for user action and for reshaping of affordances through the Internet.

This chapter is composed of four sections. In the *first section* I trace the evolution of scholarly studies of the pre-Read/Write Internet; the review of this literature is not meant to be an historiography of early Internet studies — which would be alien to the scope of the present work, and thus unavoidably incomplete: rather, although the following chapters engage in a dialogue with more recent, ethnographic approaches, my intent is to highlight aspects of earlier representations (most importantly, a distinct split between *online* and *offline*, as well as variously declined expectations of freedoms related to the extension of interpersonal communication through the Internet) whose legacy was still firmly present in the accounts of my fieldwork participants, as well as still informing contemporary public discourse. In the *second section* I focus on key concerns of the academic debates around technologies and practices close to my core research focus on computational agency on the Internet: how the transition from an early (mostly) read-only Internet to a platform enabling high levels of interactivity has been interpreted in its various aspects (social, cultural, technical, economic) in recent literature. In the *third section* of this chapter I discuss how my focus on the domain of lifeworld Internet can be reconduced to the articulation of the concept of lifeworld in phenomenological theory, and how the centrality of computation and computational agency in my analysis both complicates and questions the role of communicative action in Habermas' theory. In the *fourth section* I discuss the theoretical framework employed throughout the present study, underlying both the approach to the fieldwork and the choice of methods, as well as the analysis of the discursive materials gathered through my fieldwork.

2.1 Assembling pre–Read/Write Internet

> For the anthropologist, there is no such thing as Facebook; there is only the aggregate of its particular usages by specific populations. (Miller 2012, p153)

The relatively short history of Internet studies has been characterised by an intense unfolding of tensions alongside various directions: from the basic questions about how to delimit the object of study, what to include and what to leave out of accounts and analyses, to the disciplinary approach, to the influence of cultural framing.

Alongside the building and shaping of the Internet itself by way of technological developments, formal and informal regulations, shifting approaches to the ways the Internet is incorporated in everyday life and practices by users, the manifold approaches to Internet studies are actively contributing to the shaping and the perception of the object of study itself.

A central ambiguity, and arguably one of the reasons for the attractiveness of Internet-related practices to users and of its study to researchers, is the essential simplicity and flexibility of what the Internet is per se, before anything that happens, is experienced, created and shaped in, through, or even despite it: at its core, the Internet is essentially not much more than the suite of intangible packet-switching data transmission protocols, known as TCP/IP and first described by Cerf and Kahn (1974) and Postel (1981a), purposively designed to accommodate any practical use without mandating details at higher levels. Although it may seem naïve to adopt such a minimalistic definition for what undoubtedly is a very complex and articulated part of human life, whether directly or indirectly as infrastructure, throughout most of the world today, I find that a 'clear and unambiguous' minimal definition, such as the one proposed by Miller and Horst (2012, pp3-5) for 'the digital'[1] as foundation to the discussion of 'digital anthropology' in their edited volume (2012b), provides a solid foundation against which competing approaches and extended interpretations can be compared and critically assessed according to their ability to analyse Internet-related phenomena in specific contexts, while keeping true to the fundamental structure of the core Internet protocols.

Within the broad domain of the social sciences, as opposed to computer sciences, studies focusing on the Internet are in fact actually focusing on everything else but 'the Internet itself': individual users, groups, institutions, their practices and their interactions with networked affordances connected through the Internet protocols; normative frameworks such as laws, regulations, customs; and technologies and networks that are made possible by the 'Internet-as-a-protocol', but sit above it at multiple layers, from hidden infrastruc-

[1] "Rather than a general distinction between the digital and the analogue, we define the digital as everything that has been developed by, or can be reduced to, the binary — that is bits consisting of 0s and 1s. The development of binary code radically simplified information and communication, creating new possibilities of convergence between what were previously disparate technologies or content." (Horst and Miller 2012b, p5).

ture up to direct contact with people.

Given the 'thin' definition of the Internet outlined above and used as basic underpinning for the present study, one of the first challenges that must be addressed when conducting research related to the Internet is therefore to accurately define the domain and context of the study: this can be challenging for projects whose focus is on uses of specific technologies, or on specific contexts or practices (e.g. Andersson Schwarz 2013; Boellstorff 2008; Miller 2011; Miller and Sinanan 2014), and can be even more problematic when the core research questions are relatively open to the ways in which Internet users shape their own Internet environments and interpret their own experiences, such as in Bakardjieva 2005; boyd 2014; Miller and Slater 2000 — and to a different extent the present study.

Part of the difficulty is that 'the Internet' that emerges from different studies, even before taking into account differences in research approaches, depends largely on historical timeframes, geographical locations, material conditions and other factors that concur in shaping context and uses.

2.1.1 The early Internet as a mediated environment

Historically, as most other media, the Internet came to be gradually over time, and building upon previous iterations of similar technologies: if, for example, strictly speaking radio or TV may be considered to have come to be once the first successful broadcast of radio or TV signals was completed, in practice neither actually mattered to most of the general public as media in the domestic environment until affordable receiver units were widely available to consumers and until stations started broadcasting content relevant to people's interests. Similarly, although the earliest packet-switching wide-area network interconnecting local networks had been in operation between a few academic institutions since 1969 (the ARPANET, as outlined in Bolt Beranek and Newman Inc 1981), it wasn't until the Internet was opened up to commercial entities (US Congress 1992) and dial-up access to the Internet became available and relatively affordable for home users in the mid-1990s that the Internet started becoming relevant to households, initially blending with existing uses of non-networked home computers and related user practices.

Accordingly, when browsing some of the earlier Internet studies from only a couple of decades ago, the practices listed there may sound only vaguely familiar, from readings or movies, to today's Internet users: Rheingold 1991, for example, focuses on concepts and practices such as virtual reality, cyberspace, teledildonics, at times discussed as they are being used at the time, and often described as potential experiences that technologies may enable at decades' distance. Yet, either as experienced or imagined, those described in early literature were often practices that real (present or future) people were, or were expected to be going to be, involved in:

> You probably will *not* use erotic telepresence technology in order to have sexual experiences with machines. Thirty years from now, when portable telediddlers become ubiquitous, most people will use them to have sexual experiences with other *people*, at a distance, in combinations and configurations undreamed of by pre-cybernetic voluptuaries. (ibid., p345)

Even when not imagining Internet-connected futures (Negroponte 1995) and analysing, instead, how real users appropriate computers and the Internet in their daily lives (Turkle 1995), a pervasive idea present in most early studies is that of a distinct split, or disconnection, between the 'offline' environment and 'online' contexts. This split was evident in the foregrounding of recurrent terms such as the 'cyber-' prefix, 'online' (or 'on-line'), 'virtual' (e.g. various contributors in Bell and Kennedy 2000; Kennedy and Bell 2007; Woolgar 2002), in the centrality of issues of mediation (in turn, through a marked focus on the communicative aspects of Internet practices, within largely textual environments, under the label of computer-mediated communication or CMC: Baym 1998), of embodiment/disembodiment, and analyses of the struggles to redefine 'community' in the context of computer-mediated communication (Rheingold 1993).

As discussed in the initial Conclusions of Miller and Slater 2000, this wave of Internet studies was building on contemporary feminist works by Butler (1993) and Haraway (1991), at the exact time when the rather sudden appearance of the Internet in everyday discourse and personal experiences could

> "[s]ubstantiate postmodern claims about the increasing abstraction and depthlessness of contemporary mediated reality [...]; and poststructuralists could point to this new space in which identity could be detached from embodiment and other essentialist anchors, and indeed in which (some) people were apparently already enacting a practical, everyday deconstruction of older notions of identity." (ibid., p5)

Moreover, countercultural projects quickly appropriated the discourses of apparently boundless interconnection allowed by the Internet to imagine and enact community-focused practices online, in some cases based on geographically delimited environments (such as the WELL — the Whole Earth 'Lectronic Link — in the San Francisco area, and the Iperbole network in Bologna, Italy: cfr. Rheingold 1993; Turner 2006 for the former and Aurigi 2005 for the latter) but firmly based on the idea that the interconnectedness built and experienced by their members was representative of a larger potential. However, this was largely due to the bias of the participants in these early communities: Curran (2012) highlights how similar themes and hopes around how 'the internet would bond the world in growing amity' (Curran 2012, p8) were popular not only among activists but also theorists well into the 2000s, whereas since the earliest days of the Internet several factors have contributed to making these interconnection hopes unrealistic: most notably, inequalities in wealth and opportunities, language and cultural barriers and control operated by authoritarian governments (ibid., pp8–12).

However, the cultural climate may account only partially for the distinct 'pre-2000s' focus of Internet discourse (and Internet studies): early users were enacting and relating experiences of a split between their 'offline' and 'online' lives also because their contacts online could often only be (or mainly be) other people they did not and could not easily know in person: users of early pre-Internet and Internet online communities were relatively few and far apart, which practically made it impossible for most of them to experience the Internet as the underlying fabric weaving together established kinship ties (Madianou and Miller 2012; Miller and Sinanan 2014) or friendship and peer groups (boyd 2014).

Accordingly, the textual dimension accounted for a large part of early users' experiences online: not only in terms of physical representation of information at a time when Internet connections would not allow for reliable, high quality transmission of recorded or realtime video or audio that could enable the closeness experienced by today's users of VoIP or webcam, but also — and more importantly — through the participation in usenet forums (Baym 1995), MUDs (Bromberg 1996; Ito 1997) and various forms of online communities (Baym 1998), which — except for those linking users already known to each other in local communities such as the cases mentioned above — constituted a textual environment where people would interact with others behind nicknames and avatars which marked their identity online as distinct from their offline one.

Often offline identity and online ones would still be linked (except in the case of anonymous or pseudonymous participation in online communities), yet the perceived lack of sensorial cues while interacting online, as opposed to in person, constituted a distinct change for most users. For some (Rheingold 1993; Turkle 1995) this was seen as potentially constituting an opportunity for reinterpreting the ways in which people were seeing themselves and building relationships with others, though other studies employing social psychology techniques and studying computer mediated communication in the workplace also found the opposite — that the difference in cues available could also reinforce existing roles and interpersonal relationship patterns (Lea and Spears 1991; Marvin 1995). Turkle (2011) recently revisited her earlier positions, although the core thesis of her later critique, the increasing inclination towards replacing 'real' relationships with ones mediated through social networks, smartphones and other Internet-connected devices, is in turn further critiqued by Miller and Sinanan (2014), who highlight how for anthropologists there is no such thing as a 'real', pristine or authentic act of communication as opposed to a highly mediated one, as even direct communication in any culture is always imbued in a complex and thick network of values, customs, beliefs and other cultural structures that influence the forms of people's communication.

2.1.2 The ethnographic turn in Internet studies

If the experiences and accounts of early Internet users and theorists outlined in the previous section were influenced by the trail of existing narratives at the time of initial widespread availability of domestic Internet connections, once home Internet connections started becoming relatively widespread at least in western countries, these narratives, percolated from early user circles and from the scholarly environment to mainstream media, constituted the initial backdrop against which a wider public of users started assembling their own personal experiences of the Internet. Silverstone and Hirsch (1992) note how possible uses of new technologies do not become instantly evident to users: rather, they emerge over time through users' interactions with technologies within their specific context. In the case of the Internet, as users started spending more time online, they tried to figure out 'what to do' with the new possibilities becoming available directly in their homes through dial-up connections to the Internet. Feenberg (1995), through his theory of instrumentalization, similarly highlights how new affordances are interpreted and repurposed by users according to pre-existing needs and in the light of pressing problems they may be trying to solve.

Accordingly, as the Internet progressively started becoming more part of everyday life rather than a communication environment partially disconnected from it, scholarly research turned its focus more closely on people interacting with Internet affordances in their everyday life — rather than on the cyber and virtual worlds that had been the subject of much early research — and to include in the analysis the 'offline context' in which users' experiences of the Internet were taking shape.

Miller and Slater 2000 and Bakardjieva 2005 are among the first extensive ethnographic studies of Internet users focused on understanding how people assemble their own experiences of the Internet, starting from their situated context (and hence from a broader overview of their lives and of temporal, cultural and geographic context) rather than from 'what they do online'. As Miller and Slater suggest while analysing the 'focus on virtuality or separateness as the defining feature of the Internet' (Miller and Slater 2000, p5) of previous studies:

> [i]f you want to get to the Internet, don't start from there. The present study obviously starts from the opposite assumption, that we need to treat Internet media as continuous with and embedded in other social spaces, that they happen within mundane social structures and relations that they may transform but that they cannot escape into a self-enclosed cyberian apartness. (ibid., p5)

These ethnographic studies, as well as the many that followed (Madianou and Miller 2012; Miller 2011; Miller and Sinanan 2014; Slater 2013 amongst the most recent and relevant to my study, although already Hine 2000 — notwithstanding the somewhat ambivalent title — was building on ethnographic research through a sensitivity similar to those of later studies), marked a distinct turn in scholarly approach but were also timely: they examined the practices of users within specific fieldwork contexts at the time that a rapidly growing cohort of users was finally able to connect to the Internet and to personally try to make sense of how the ability to exchange information quickly could be put to use within their individual, situated contexts.

In a network of mutual influence, arguably without any univocal and unidirectional nor linear causation, user experiences, mainstream discourses, academic discourses, commercial interests and technological developments contributed to a swift reshaping of each other in the years of frantic development (Crain 2014) of Internet advertising and commercial ventures leading to the year 2000 'dotcom crash': rather suddenly, 'the Internet' became part of the fabric of daily life — and as such the earlier paradigm of disconnected realities was challenged, often implicitly, as something not relevant anymore to the way in which people were appropriating, shaping and interpreting the Internet as one of the many traits of the material culture of everyday life.

2.2 Assembling Read/Write Internet

In what follows I will analyse the academic literature relevant to the multiple aspects under which Read/Write Internet can be examined; to complement and extend the following discussion of concepts and theories, a more technical discussion, focused on the materiality of technologies, underlying economic structures and surrounding discourse of the Read/Write Internet is presented instead in Chapter 4.

2.2.1 Unfolding the Web 2.0 narrative

> The World Wide Web in 2003 is beginning to fulfil the hopes that Tim Berners-Lee had for it over 10 years ago when he created it. The web was never just supposed to be a one-way publishing system, but the first decade of the web has been dominated by a tool which has been read-only — the web browser. The goal now is to convert the web into a two-way system. Ordinary people should be able to write to the web, just as easily as they can browse and read it. (Macmanus 2003)

Although the Internet at large had been a two-way medium since its early times (online communities, forums, MOOs are just a few example of early interactive environments), the web, specifically, although designed to be a read/write medium, only started to be used consistently for more than personal homepages, 'company brochure' and ecommerce sites in the early 2000s: statements such as the one above by Macmanus and the oft-cited 'What is Web 2.0 — Design Patterns and Business Models for the Next Generation of Software' blog post by O'Reilly (2005a) started capturing essential lines of the ongoing transition to a two-way web as this was happening and becoming increasingly relevant to the general public.

More specifically, in the context of the present dissertation, these traits also outline a way of using the web (and, increasingly, as users started adopting smartphones and tablets, the Internet at large) that quickly shifted from (mostly) read-only to read/write patterns. It is important to highlight that *read/write* is not meant here to simplistically denote the ability to publish content online: the quote from Macmanus 2003 at the beginning of this section, as well as Lawson 2005, focus on the browser environment as read-only, in comparison to Berners-Lee's original web navigator, which was designed also as an editor of web pages:

> The idea was that anybody who used the web would have a space where they could write and so the first browser was an editor, it was a writer as well as a reader. Every person who used the web had the ability to write something. It was very easy to make a new web page and comment on what somebody else had written, which is very much what blogging is about. (ibid.)

However, although not as accessible to a wide public, the creation of carefully handcrafted personal websites was a popular pastime way before blogging platforms became available, as testified by the huge success of hosting platforms for personal websites such as GeoCities in the second half of the 1990s (Roberts 2000).

Throughout this thesis, instead, Read/Write is used to capture a much broader set of user attitudes towards ways of incorporating the Internet in their everyday life, starting indeed from the ability to publish content online, but declining this into multiple forms, spanning over new ways to access media, contribution to projects relying on user-generated content, interaction over social media and within social network sites. Furthermore, although the domains just listed reference technological affordances such as social media web applications, my focus is in fact on the 'cultural shift', and what it means to individual users, underlying practices that became increasingly 'normal' as the Read/Write Internet developed, similarly to Jenkins' (2008) note on convergence culture:

> I will argue here against the idea that convergence should be understood primarily as a technological process bringing together multiple media functions within the same devices. Instead, convergence represents a cultural shift as consumers are encouraged to seek out new information and make connections among dispersed media content. This book is about the work — and play — spectators perform in the new media system. (2008, p3)

In the following sections I will outline elements of the scholarly debate about how — even when phenomena are named differently — a Read/Write Internet and a related cultural shift have been taking shape.

2.2.2 Read/Write culture

One of the first domains in which such a cultural shift became apparent is that of cultural artifacts: thanks to digitization of content, accessing and sharing music, video and text became much easier than in the pre-digital era, leading to new ways of interacting with culture.

For the wider public of Internet users, finding music online through an user-friendly interface started with Napster in 1999, as outlined by Lessig (2002):

> No doubt the most famous story of musical "innovation" has been the explosion called Napster — a technology simplifying file sharing for MP3 files. [...] Fanning and Parker's idea was just this: Individuals had music stored on their computers [...] Others would want copies of that music. Fanning devised a way to engineer a "meeting" between the copies and those wanting them. (Lessig 2002, p130)

As outlined in the quote above, Napster didn't represent an innovation in terms of making it possible to 'write' — to make music available on the Internet from one's hard drive: making perfect digital copies of music from audio CDs and video from DVDs to a computer and sharing them was already possible and widespread at the time of Napster's launch (ibid., chapter 11, which also discusses the technical restrictions to digital copies (DRM) backed by laws such as the USA's DMCA, Digital Millennium Copyright Act). The real innovation of Napster, instead, was to connect users who had some specific music albums with those who were looking for them. In fact Napster, as the peer-to-peer alternatives such as the bittorrent infrastructure, which became popular after Napster's shutdown in 2002 by court order, was not allowing users to upload copies of MP3 files to its own servers, but merely sharing 'match-making' information between users.

Vaidhyanathan (2004) extends Lessig's analysis of the technical and legal details of digital copying and filesharing by providing a broader overview of the cultural changes taking shape within the ongoing tension between the extremes of total control ('oligarchy') and total deregulation ('anarchy'), represented respectively by the incumbent media industries on one side and by users on the other, with fringe musicians and artists variously inbetween, according to their willingness to relinquish control over copying of their works in order to engage in different ways with their audiences, for example by promoting fan remixes.

Andersson Schwarz (2013) develops a nuanced understanding of filesharers' motivations, highlighting how access to music, films, videogames and ebooks via p2p networks 'more often than not [...] hinges on the individual end user's desire to acquire entertainment and to maximise both pleasure and efficiency' (Andersson Schwarz 2013, p2).

Andersson's analysis highlights how in the case of p2p filesharing two facets of Read/Write Internet are conflated: on one hand, the technical side (the bittorrent protocol, bittorrent apps and tracker sites) quickly became part of a Read/Write infrastructure now often taken for granted by users, and often pragmatically used to overcome the perceived limitations of legal media distribution websites provided by the incumbent media industries: these are often seen as failing to provide the convenience of access and the immediate availability of content offered instead by the p2p networks, for example by releasing episodes of TV series only months after they have been broadcast, and only in some geographical areas, frustrating international publics and migrants who wish to watch TV series from their own country while abroad.

On the other hand, the social and cultural sides of Read/Write Internet inherent in user engagement in p2p filesharing can be best understood at the light of Chris Anderson's concept of the 'long tail' (2004): his focus there is on the ability to match consumers' desire for niche products through the convenience of digital duplication from a producer/distributor's perspective, although arguably the very ability to match niche requests (which by itself would be a traditional 'read-only' provision of content for consumption, only in digital format) presupposes the opposite ability, that is for individual users with even unique requests to make their needs manifest (thus implying a read/write use of the Internet), as well as the possibility for individuals to get to know more easily, through peer engagement, lesser-known artists whose popularity would have otherwise been mostly dependent on media companies' decisions on which artists to give visibility to.

2.2.3 User–generated content and engagement

Another cultural shift precipitated by the Read/Write Internet is the direct engagement of users in public cultural creations, either individually (through curation of blogs or personal collections of photos, videos, texts, illustrations) or as part of groups (such as by contributing and reviewing content on collaborative websites such as Wikipedia or OpenStreetMap, or by taking part in fan-generated narratives based on popular cultural productions): the various aspects of this shift, usually analysed under the label of *user-generated content*, are outlined in the following paragraphs.

Jenkins (2008), as part of a broader discussion about convergence culture, examines through several case studies the motivations of media consumers as they turn from passive viewers or readers to active contributors to spoilers forums and fan-generated cultural remixes, as well as analysing the media industries' varied ability to understand the role of user engagement, either fostering it as part of their efforts to curate their brand marketing or countering it — and often facing backlashes as a consequence.

In countless practical articulations of the Web 2.0 focus on harnessing collective intelligence, user engagement with mainstream media content is made possible or facilitated by the Internet: whereas early Internet users could already congregate in online forums and communities to discuss fan fiction and other shared interests around mass media productions, Jenkins highlights how media convergence and the availability of multiple ways to access content, comment on it and remix it (extending the Web 2.0's trait of 'software above the level of a single device' to content more generally) make user engagement richer, more immediate and publicly visible, while on the other hand allowing brands to catalyze user-generated content to amplify the visibility of their cultural productions.

Conversely, Keen (2007) argues that the ability for anyone to 'write' anything on the

Internet constitutes a mismatch against the ability to produce meaningful and reliable content: his critique of user-generated content in a Web 2.0 context is focused on what he perceives as a progressive dilution of the aesthetics and civic discourse of the Western world due to the increasing availability of vast amounts of amateur content lacking the vetting that in his view guaranteed the quality of content in pre-Read/Write Internet media flows.

Jenkins, Ford and Green (2013) further extend Jenkins' own (2008) analysis by articulating the nexus of often contrasting interests of media industries and individual content creators against the backdrop of the increasing popularity of social media: whereas the first shift to a read/write user-generated content entailed more linear relationships between users/fans and between these and the media industries, the pervasiveness of social media and social network sites complicates these relationships, and more importantly makes it more problematic to clearly trace agency and power within the complex networks of content, media and users involved.

These often blurred and volatile reconfigurations of agency and power following reconfigurations of the ways in which users engage with content on a Read/Write Internet are also examined by Feenberg and contributors (Feenberg and Friesen 2012) within the theoretical horizon of critical theory of technology; whereas user-generated content is mostly seen within the Web 2.0 paradigm as a way to channel the fragmented value of users' collective intelligence (Lévy 1997) towards broader cultural products such as Wikipedia and OpenStreetMap, critical theory of technology focuses on users' intentionality, sense of purpose and quest for meaning, which collectively inform a political agenda. Content, in this view, is subordinate to action, be it discussion, engagement with environmental or political concerns, or what Bakardjieva (2012) calls subactivism:

> [...] a kind of pre-politics that opens spaces for agency in relation to institutions such as the medical system, government agencies, and schools. (Feenberg 2012, p15)

According to Feenberg (2012), Read/Write is a core trait of the Internet, understood as the ability to enable political engagement:

> The Internet has the power to put those involved in [...] technically mediated environments in contact with each other. What is most innovative and politically significant about the Internet is its capacity to support collective

> reflection on participant interests. (Feenberg 2012, p15)

As in the cases of fan engagement discussed above, this ability to foster collective reflection is not a distinctive trait of the Read/Write Internet per se in a technical determinist sense: to unfold the conditions for this to happen, Feenberg traces an evolutive path of agency on the Internet and of its political implications across three stages, which, albeit not mutually exclusive nor strictly successive, nevertheless capture a trajectory of evolution from a read-only model ('The Information Model'), through an hybrid mode mainly fuelled by the pre-2000 high hopes in the commercial potential of a fast-spreading Internet ('The Consumption Model'), to a 'Community Model' which predates the Web 2.0 but is increasingly reshaped by Read/Write technologies, as conveyed by the following quote which highlights both the importance of two-way, or Read/Write, interaction and of the (technical) ability to reconfigure the medium without the intervention of centralized powers:

> The essence of the community model is reciprocity. Each participant is both reader or viewer and publisher. [...] It must be possible to introduce innovative designs for new forms of association without passing through bureaucratic or commercial gatekeepers. (ibid., p12)

2.2.4 Critiques of Web 2.0

Besides Keen's critical assessment of the Web 2.0 outlined above, others have questioned the Web 2.0 discourse along several dimensions. Han (2011) identifies three main strands of critique: he includes Keen's position (identified as 'elitist', Han 2011, p98) and further adds a 'non-neutral' strand (ibid., p91) and a 'leftist' strand (ibid., p103).

The 'non-neutral' critique is exemplified by Han through the works of Carr (2008; 2011) and Lanier (2010, though his later 2013, which was published after Han's, further elaborates Lanier's position) and invokes the conviction that the modes of online interaction enabled by the Web 2.0 actually have 'lasting effects on what it means to be human' (Han 2011, p91). These two authors' polemic objectives, however, are in fact quite distinct in terms of specific aim.

Carr draws on psychology and brain sciences studies to argue that the way of interacting with content on the contemporary Web is favouring 'skim-reading' and a fragmentation of attention which extends beyond the time spent online, causing longer-lasting effects in general attention patterns. Lanier's critique, instead, is similarly concerned with wider effects of the sudden pervasive uptake of practices shaped by the current Web, but focuses specifically on the implications on the interface between machines and humans of what he identifies as the 'computationalism' ideology of Web 2.0: "This ideology, promotes radical freedom on the surface of the Web, but that freedom, ironically, is more for machines than people. Nevertheless, it is sometimes referred to as 'open culture'" (Lanier 2010, p3, in Han 2011). Although controversial, both Carr's and Lanier's critiques point to still under-analysed broader transformations of human behaviour, social expectations and cultural processes which resonate within my fieldwork materials. Longer term research is certainly needed to establish more grounded causation links and general patterns, however: Carr's and Lanier's theses could in turn be subject to a similar critique as the one raised by Miller and Sinanan to Turkle's (2011) arguments about a waning of the supposed immediacy of much pre-Web 2.0 interpersonal communication, and it could be argued that similar arguments raised by fieldwork participants could at the same time reflect a valid perception of changes happening to their own rapport with culture and the world, while also being shaped by the 'non-neutral' discourse routinely exposed in mainstream media.

Han's analysis of the 'leftist' strand of critique to Web 2.0 is focused on the work of Geert Lovink, who — according to Han — is equally concerned with the fact that "the sociality of Web 2.0 depends on corporate technologies" (ibid., p105) and that the political implications of any democratic potential assumed to be made possible by the Read/Write character of Web 2.0 is illusory because of the "temporary nature of the new wave of social technologies. [...]": Lovink and his co-authors "have very little faith that Web 2.0 will form lasting *institutions*" (ibid., ibid.). At stake in this strand of critique, which Han links to the Frankfurt School and to Herbert Marcuse in particular, is essentially the concern that any mild democratization of some aspects of public life thanks to progressive practices supported by the Web 2.0 ends up in fact relegating more systemic concerns about the functioning of modern democracies to the back of a thin layer of convenient, and ultimately largely ineffectual, access to some form of direct participation in public life.

Lovink's theoretical concerns have informed the creation of activist networks focused on the imagining and development of alternatives to the 'corporate technologies' whose hegemonic role in Web 2.0 practices is denounced by Lovink: the engagement of some of my fieldwork participants with such ideas and practical alternatives to corporate technologies is further analysed in the following empirical chapters.

A different strand of critique of Web 2.0, highlighting past failures with the pragmatic aim to outline ways in which media companies could better serve the interests of users is offered by Jenkins, Ford and Green (2013): they focus on the user-generated content and participatory culture aspects of Web 2.0, arguing that media companies have too often been narrowly focused on 'harvesting' the benefits of user engagement, under pressure from investors and other stakeholders, while ignoring the moral economy upon which tacit agreements with users rely: as users contribute content, they expect to be able to continue to build upon it and to enjoy it on their own terms, rather than seeing it become exclusive property of the commercial entities which sought to promote community engagement only to sequester its fruits into tightly controlled walled gardens. Amongst other scholars who focus on this strand of critique, Petersen (2008) usefully highlights tensions inherent in the unfolding of Web 2.0, which he traces back to a fundamental 'messy relationship' between users and corporations: exploitation of free (user) labour to foster corporate aims can be as fundamentally hardwired into the technical design of Web 2.0 apps as to make it impossible for users to meaningfully contribute content freely and curate it out of personal interest through web apps (such as the Flickr photo sharing platform, which is one of the focuses of Petersen's analysis) without their actions becoming part of systemic exploitation due to the constraints enforced by even relatively open platforms with seemingly open APIs:

> Flickr is very active in promoting themselves by opening up their API so people will code small applications that improve the design and usability of Flickr, while at the same time helping Flickr in promoting it as the cool site to be at. There are limitations though as to how open the API is; the openness stops short of enabling migration. It could be an option to migrate if you could take the entirety of the context data with you. (Petersen 2008)

Petersen interestingly calls for 'a theory of labor that is able to map both exploitation and free labor, along with considering the value using these sites creates for their users' (ibid.), although his analysis rests on the fundamental opposition between corporate capital and user labour, hence focusing on the trait of Web 2.0 as enabler of generation and sharing of content, leaving out of the analysis the potential role of reassembling of the Web 2.0 infrastructure through the labour of developers inside and outside of corporate settings, complicating the assumption of a insurmountable chasm between corporations and users within a capitalist social and technical organization.

2.3 Internet and lifeworld

The use of the *lifeworld Internet* label, as discussed in the Introduction, is motivated by the need to clearly identify the domain of user practices investigated throughout the present work, rather than by the intention to introduce a fully original concept. It is however useful here to articulate how and why this concept is used, briefly tracing its connections to core theoretical references, and how my focus on computation and computational agency allows to deconstruct spaces for political contestation that are not fully captured by theoretical approaches focused on the communicative—rather than computational—traits of the Internet.

The relevant formulations of the concept of lifeworld in Western philosophy can be originally reconduced to Husserl's work: *Lebenswelt* is complexly articulated (Husserl 1970) as both the personal horizon of beliefs through which the outside world is structured by the individual, and as the intersubjective systems of meanings that allow communities of individuals to make sense of shared experiences and to translate between different communities' own languages. Whereas the modern heritage of the concept of lifeworld has its roots in Husserl's focus on subjective experience and consciousness, as opposed to 'theoretical–logical substruction' (ibid., p127), Habermas' focus on communication and his articulation of the distinction between lifeworld and system (Habermas 1987) constitute a closer foundation to the conceptualization of lifeworld as used here within the domain of lifeworld Internet.

Drawing on the phenomenological approaches of Husserl (1970) and of Schutz and Luckmann (1973), Habermas highlights the role of language and of communicative competence in processes of cultural reproduction, social integration and socialization within the structural components of lifeworld (culture, society, personality: Habermas 1987, ch6.1). Although developed with a focus on pre-Internet media, Habermas' theorization of lifeworld can usefully help to articulate on one hand the domain of practices that pertain to an everyday, personal use of the Internet (where the same processes of cultural reproduction and sense making, social integration and socialization can clearly be identified, especially when analyzing user practices on social network sites) and on the other hand the tensions with systemic forces that shape and constrain the technical structure of mainstream Internet.

Likewise, the role of communicative competence as introduced by Habermas can further be deconstructed through the complex interrelation of personal expression in a polymedia (Madianou and Miller 2013) stage of the Internet—involving modes of expression and relationship with publics vastly different than those of pre-Internet personal communication—with the technical materiality of the affordances through which this communication is performed, managed, enjoyed and consumed. Interestingly, however, Habermas' relatively scant and late engagement with the Internet (Habermas 2006a,b), by focusing on communication happening through the Internet rather than on the reshaping of individuals' lifeworlds within the small–scale domain of everyday life, seems to be missing the potential inherent in his own earlier theory to highlight the weakening of communicative competence through the influence of systemic structures (such as the computational infrastructure and its corporate agendas that I examine in the following substantive chapters), as well as the democratic potential of alternative rationalizations of the computational infrastructure. This allows Keen, for example, to read and appropriate Habermas' comments within his own 'elitist' critique of Web 2.0 (Keen 2006), whose limitations I have discussed in the previous section, in contrast to Feenberg's critical theory of technology.

Therefore, as suggested in the Introduction through the mentioning of Marcus' (1995) work, whereas my analytical framing of lifeworld and of its contested relationship with the colonization of everyday life operated through system–level technostructures references the phenomenological tradition, the actual research processes and methods through which the domain of lifeworld Internet is explored and hacker contestations are followed relate more closely to the broad ethnographic tradition that since the 'ethnographic turn' of

Internet scholarship has been focusing on the practices and representations of individual users and of hackers involved in the use and in the assembling of the Internet, as articulated by Bakardjieva (2005, pp38–43) with reference to the work of Schutz and Luckmann (1973).

Although in the present work I strive to develop a critique of determinist technological rationality—which could be considered as an instance of 'critique of functionalist reason' (Habermas 1987)—within the specific domain of lifeworld Internet, a core distinction from communicative theories of the public sphere such as Habermas' one is here constituted by the central role that computation and computational agency have in the articulation of the negotiations operated by users and hackers, further deconstructing the role of communicative competence by highlighting how this—in the case of the Internet—is itself ultimately intimately influenced by the power relations that shape the computational infrastructure over which communication itself is performed.

2.4 Conceptualising power struggles around the Read/Write Internet

In the previous sections I outlined several dimensions alongside which the experiences and practices of Internet users are necessarily to be questioned and put in context; more expansive analysis of these and other dimensions similarly affecting the shaping of 'the Internet' as visible to end users will be introduced in the following empirical chapters.

At the light of these large-scale and intricately interdependent issues and sources of power operating at different layers of the technical, political and economic infrastructure of the Internet, my research tries to contextualise user actions, experiences and accounts within a much broader landscape which includes both human and non-human actors: in this final section I outline concepts and literature which are relevant to the ways in which my research tries to make sense of this complex landscape.

As discussed in the first section of this chapter, early representations of the Internet often included expectations of improvements of people's lives — from the intimately personal to the social and political spheres — thanks to the new opportunities opened up by the Internet. Although, as outlined above, there were dissenting voices, a substantial technological determinism was underlying these initial hopes; this is most evident in the ICT for development (ICT4D) discourse, within which huge amounts of money from mostly Western aid programmes were poured into ICT-focused programmes (again, often with a Western-sanctioned view of what ICTs and media meant) in the global South, with limited understanding of the local material conditions and the often radically different frameworks within which beneficiaries of these programmes were operating, with the underlying assumption that deployment of technology would necessarily bring improvements (Slater 2013 develops a comprehensive overview of these issues within the broader context of challenging the North-dominated view of the nexus of new media, development and globalization).

My research, necessarily, moves from the opposite direction, that of social constructivism. The approach most relevant to my research is that of Social Construction of Technology (SCOT), first outlined in Pinch and Bijker 1984 and further expanded upon in Bijker, Hughes, et al. 1987: the SCOT approach questions panglossian views of technological development, highlighting how technological configurations are always the result of complex interactions and negotiations of different groups, knowledges and interests, which progressively lead to the definition of technological artifacts: these could have been developing in several other ways under different conditions, rather than according to an inevitable and unidirectionally linear technological rationality. Social actors, therefore, are understood as primarily having the capability to influence the construction of technology, by exploiting what Pinch and Bijker term *interpretative flexibility* (Pinch and Bijker 1984, p411), the openness of technological artifacts to being interpreted in different ways by different stakeholders. To explain the process of negotiation leading to a concrete outcome out of the several possible, the authors also introduce the concept of *relevant social groups* — formal or informal groups or organizations involved in the negotiations around a technological development, with the key requirement 'that all members of a certain social group share the same set of meanings, attached to a specific artefact' (ibid., p414).

Bakardjieva (2005, p11) outlines key criticisms to which the SCOT approach has been exposed:

- the central role played by relevant social groups can easily obfuscate and ignore the voices of those who (either by being individuals, such as end users not connected in larger groups, or by being part of minority or underrepresented groups) are not able to act as a relevant social group and therefore to influence the shaping of technologies being developed
- the role of forces remote from the processes of negotiation but with a strong influence on it is not properly taken into account: only the voices of actors immediately involved are thought to be actively shaping technologies, therefore excluding the layers of social structure which underpin these actors' worldviews, assumptions and agendas

These concerns are highly relevant to my own research, which on one hand follows individual users' struggles with technology, and on the other aims to include in the analysis the complex network of forces which operate throughout the infrastructure of the Internet and throughout society at large.

Whereas users — especially users of Internet technologies — could in theory gather in groups (e.g. via online forums) to discuss their needs and to make them visible, thereby acting as a relevant social group, in practice the imbalance of power in favour of formal, or well-funded groups is hard to overcome: a context in which group engagement has been shown to be potentially successful is that of patient networks (e.g. Laurance, Henderson, et al. 2014), although arguably in the rather different context of medical health care and peer support rather than shaping of technologies.

While building on a social constructivist perspective, my research tries to address these limitations of the SCOT approach through complementary strategies. In order to account properly for individual users' voices, I try to not only take each participant's account as a substantive counterpoint to prevailing narratives of 'what the Internet is' as a series of black-boxed artifacts interconnected by the Internet protocols, but I build on the preoccupations and methods of the material culture approach to trace the accounts of participants even further and to try to understand their experiences at the light of the broader context of their lives, even — and perhaps especially — the areas that are not directly relevant to their use of the Internet. This emerged from realizations through my fieldwork, when students would try to make sense of their experiences of the Internet through some apparently distant and unrelated experience, which however in their account did shine a different light on their practices online.

The approach I follow is perhaps best synthesized by Miller (2009):

> Such a perspective seems properly described as *material culture* since it implies that much of what makes us what we are exists, not through our consciousness or body, but as an exterior environment that habituates and prompts us. (Miller 2009, pp50–51)

Miller's own approach to understanding 'what people are' is perhaps best exemplified through the portraits of his monograph *The comfort of things* (2008), although — specifically related to the Internet — Miller and Slater 2000 was already fully tracing the Internet practices of Trinidadian users starting from the 'exterior environment' to make sense of their experiences and accounts. Similarly, the following ethnographic studies already mentioned in the previous sections (Madianou and Miller 2012; Miller 2011; Miller and Sinanan 2014; Slater 2013) successfully deployed a material culture understanding of the respective fields to provide much richer and grounded accounts than what would have been possible by simply observing Internet users while they interacted through computers or other devices.

While per se a material culture approach is obviously not enough to give voice to individual users and independent developers as part of relevant social groups, my argument is that only by understanding each user's and developer's experience and motivations in the broader context of their lives and material conditions an ethnographer (including those who are employed at companies developing technologies related to the Internet, in a long tradition stretching back to the 1970s — see Suchman 2013) can gain a fuller understanding of the fragmented public of users, with the aim to inform a more just development of technologies which takes into account the needs and sense-making processes of individual users even when they are not able to act as a proper relevant social group.

To try to address both the limitations of SCOT highlighted above with reference to my fieldwork material, furthermore, I employ the framework of critical theory of technology, which Andrew Feenberg has been developing throughout the past two decades. Building upon the early theories of technology of Martin Heidegger and on the social and political analysis of Marcuse (1955; 1964), who had highlighted how technology in advanced industrial societies had already become a powerful means of domination and reproduction of imbalances of power, Feenberg's theory engages with the social constructivist tradition, while trying to extend its reach with the explicit objective of enabling democratic and inclusive shaping of technology.

Similarly to material culture studies although under different terminology, Feenberg's analysis (1999) includes the lifeworld of users as a core component of how technology is experienced in practice by individuals, thereby giving central importance to the highly specific ways in which users make sense of technologies, artifacts and affordances beyond the realm of their pure technical structure. Accordingly, his theory is concerned with issues of agency that extend beyond the immediate reach of what is discussed and decided by the developers of technologies and what is experienced by the end users once they receive, or get in contact with, technological artifacts as black-boxes:

> Critical theory of technology departs from mainstream STS in treating such technological worlds as terrains of struggle on which hegemonic forces express themselves through specific design strategies in opposition to subordinate groups that are more or less successful in influencing the future form of the artifacts with which they are engaged. (Feenberg 2012, pp3–4)

Although centrally preoccupied with the role of human actors involved in the design and use of technology, critical theory of technology strives to identify the sources and layers of power that are hidden behind technologies as experienced by users, as a necessary precondition for the process of *democratic rationalization* (1999) — the appropriation of new technologies by users with the aim of subverting the structures of power embedded in the current, hegemonic technical code.

Through Feenberg's own and co-authored contributions in the recent Feenberg and Friesen 2012, earlier traits of his critical theory of technology are further developed in the specific context of the Internet, outlining an articulated critical theory of the Internet to which my own research relates, both methodologically (through the choice of contexts investigated in my fieldwork, as discussed in the next chapter) and by framing the various alliances between relevant social groups analyzed in the later empirical chapters as constitutive of struggles to reconfigure agency through a 'community model' of configuration of the Internet (Feenberg 2012, pp12–14), which in my own work also extends to human and non-human actants that Feenberg's earlier research could not have included for historical reasons (namely, the computational infrastructure emerging through the computational turn of the Internet discussed in Chapter 4 and the recursive public of hackers and their labour, around which Chapters 7 and 8 are focused).

Chapter 3

Research methods: choices and challenges

3.1 Introduction

In this chapter I outline the methodological issues related to my research and I discuss how the aim of deconstructing computational agency was achieved through the choice of methods, ethnographic sites and discursive materials analyzed, and how the different strategies employed throughout my project were connected to each other and constantly reconduced to the main research question.

The central concern of this thesis is the role of *computational agency* within the domain of *lifeworld Internet*: the sphere of practices and technical affordances that involve the use of the Internet throughout the everyday lives of common users. Whereas computation itself may strictly be considered, as a mathematical abstraction, entirely non-material[2], its consequences are almost invariably material, whether physically so (e.g. a turnstile being unlocked — or refusing to do so — when presented with a contactless fare card) or in the basic sense of causing a material change in the state of things (e.g. a ticket price being debited to the account associated with the fare card presented at the turnstile, to extend the same example). Computation, however, can be considered highly material also when looking at the ways in which it is developed, configured and performed through human labour: architecture of underlying networks, choices of software engineering, where and how code is run and data is stored, how representations and agendas of institutions, developers and users are negotiated into the practical implementations of software are only some of the factors that materially shape computation, in turn inscribing in technology values and configurations of power that foster specific kinds of computational agency while limiting others. This is, however, only an abstract description of a kind of politics of computational agency that is often overlooked because of its lesser visibility (due to its largely infrastructural and 'behind-the-scenes' character) than the politics surrounding the effects of computation, while being similarly 'messy' — both in the ways it operates and in how it can be investigated. But empirical research must start somewhere, and in the following I illustrate how the choice of sites for the ethnographic fieldwork informing the empirical chapters of the dissertation follows the focus of enquiry.

Through the initial fieldwork — an autoethnographic case study centered on the development of a Web 2.0 application for the students of Goldsmiths university in London — my aim was to explore developer and user agency as they would be configured within the 'content mashups' narrative of early Web 2.0 (2005-2007), related to the sphere of individual lifeworld. The specific nexus of Web 2.0 strategies and technologies, educational setting and technological mediation of everyday life constituted a particularly interesting and promising fieldwork focus at the time, in comparison with Web 2.0 research contexts such as peer-production of content, online activism, etc., which were already more widely explored. In practice, the case study highlighted a much more nuanced network of computational agency, which involved other classes of actors beyond users and myself as developer: institutional interpretations of roles of knowledge and technology; values and

[2] Turing's '*a-machines*' (automatic machines, now commonly referred to as 'Turing machines') were first described by Alan Turing (1936) as mathematical abstractions; although these machines are apparently simple, Turing showed that '[i]t is possible to invent a single machine which can be used to compute any computable sequence' (Turing 1936, §6). Through the introduction of these all-computing machines ('universal computing machines'), Turing effectively lay the mathematical foundations to the practical implementations of general-purpose computers that today constitute an increasingly pervasive presence in users' everyday lives, from personal computers to smartphones. Interestingly, whereas they are presented as mathematical abstractions, these machines were introduced in Turing's seminal paper through a lexicon that referenced material affordances and operations ('machine', 'tape', 'writing', 'scanning'): although this was arguably done to clarify the introduction of new concepts at a time when computation could barely be more than an abstraction, this choice of lexicon was an uncanny premonition of the importance that material factors would have, much later on, in the configuration of computational agency, as discussed throughout the empirical chapters of this dissertation.

norms inscribed in software libraries and software engineering best practices; as well as shifting cultural representations of what the Internet and Web 2.0 were becoming — in the broader contemporary discourse — at the time of my fieldwork.

The choice of such field therefore allowed to identify areas of enquiry that transcended the immediate spatial and temporal limits of the specific case study, and which I revisited in the successive stages of my research. My return to the same local context in 2010-2011 was aimed at understanding students' practices and struggles with lifeworld Internet in the changed context of widespread use of SNSes such as Facebook, through ethnographic fieldwork based on in-depth interviews: this new stage of fieldwork allowed me to capture, for a second time, much more than I had originally expected. This was achieved by tracing back the accounts of the students I interviewed to the power relations that continued to shape the development of mainstream Internet through the mediation of pervasive computation, on which the operativeness of Internet affordances had already started to rely much more decisively than at any earlier stage of development of mainstream Internet (cfr. Chapter 4, section on the 'computational turn' of the Internet). My later fieldwork (2010-2015) was focused on understanding the discourses and practices of a new global class of hackers which is here seen to be actively developing public counter-narratives and alternative rationalizations to the hegemonic genres and technical form of lifeworld Internet: accordingly, this stage of research was carried out across a multiplicity of locations and mainly from remote[3], by analyzing the growing corpus of talks, presentations, opinion articles, blog posts and unstructured conversations through which hackers involved in alternative rationalizations have been exposing their concerns about centralized control of lifeworld Internet, as well as illustrating their work aimed at developing computational agencies that could subvert the power imbalances of mainstream Internet.

This chapter is composed of three sections. The *first section* describes the site and contexts of the initial autoethnographic case study (whose findings are discussed in Chapter 5) and of the successive ethnography of user accounts (whose findings are discussed in Chapter 6); I also discuss to which extent this research can be considered ethnographic.

[3]During the timespan of this phase of research I attended a few developer conferences and gatherings in my capacity of social researcher, but the vast majority of relevant discursive content was reviewed through audio and video recordings of live events, as well as through texts published online or in paper format.

The *second section* looks at the methodological considerations involved in the process of recruiting and interviewing students, and examines how the different levels of technical expertise encountered were taken into account and contrasted to the ideal-type of the ordinary user. The *third section* looks at the different dimensions through which I attempted to make sense of fieldwork findings: here I analyze how prior knowledge and my role as both researcher and technical expert was taken into account, and to what extent the different parts of my fieldwork were informed by autoethnographic methods. I also analyze how 'actors were followed' beyond the field, by highlighting the ways in which my later research (around which Chapters 7 and 8 are structured) investigated actors and practices further away from the local case studies, as part of a broader network of labour, knowledge and power that needs to be questioned in order to gain a better understanding of the core issue of computational agency within lifeworld Internet.

3.2 Researching computational agency: fieldwork sites and contexts

3.2.1 From a failed Web 2.0 experiment to ethnography of student accounts

This study was originally designed as an action research ethnographic project: in mid-2005 I was appointed as web developer in the small Learning Technologies centre of Goldsmiths, a university campus in South-East London, with the remit to design and develop an online system to assist students in planning their personal development through their time at university; the web application would help students set their personal development goals, record their achievements and progress and any issues experienced, and it would help them reflect critically on their own development.

I had actually applied for the post with the hope to use the project I would be developing as the core of my fieldwork: my successful job interview presentation highlighted how a wide variety of established web applications could be used by students to record various aspects of their development (for example, blogging platforms to record more discursive notes, photo sharing platforms to store visual elements of their plans, calendar and 'to do' apps to set and review goals and for time management); my suggested design — at the height of the popularity of web apps focused on user-generated content under the label of Web 2.0 — was to let students choose which specialised Web 2.0 applications to use, leaving to the college's platform I was proposing to develop the task of simply bringing together through light coordination parts of their content stored elsewhere, as they would deem useful and appropriate in order to review their development. My plan was to carry out ethnographic interviews and observation as part of my job, in order to inform the design and development of the platform, and at different stages undertake more fieldwork to review how students were using the platform and to develop any new features or to update, redesign or remove existing ones based on student feedback.

Unfortunately, despite extensive fieldwork with members of staff from all the departments and with a small initial set of individual students, institutional requests steered the project in a different direction, relying more on downloadable booklets, forms (designed to be printed and filled-in on paper rather than updated and stored online), exercises and other materials, rather than on the distributed platform I had envisaged. Nevertheless, the experience of negotiating multiple and often contrasting agendas, navigating the uncertain terrain of nascent Web 2.0 software engineering practices and developer tools, and struggling with a much lower familiarity of students with Web 2.0 apps than what I had assumed during my initial design, when analyzed retrospectively, constitute a complex case study of situated computational agency. The ultimate failure of my project effectively contributed to exposing more clearly to my analysis the power relations already operational through the early stages of Web 2.0 discourses.

My experience in this fieldwork site, moreover, helped me to design a second phase of my research project and gave me access to resources (contacts, information) that proved invaluable once I decided to shift the focus of my research towards exploring the reconfiguration of computational agency in lifeworld Internet at a time of fast expansion of social network sites. This second stage of research relied on students' accounts of their experience of the Internet in daily life; the highly international and mixed student population (both in terms of social/economic background, previous education, age, country of origin/home culture, religion, gender and sexual orientation) represented a unique opportunity to draw from a diverse pool of Internet users, while limiting the geographic extent of the field to a very manageable urban university campus. This aspect, as discussed later in this chapter, mattered particularly in terms of being able to interview in person students from very different backgrounds rather than relying on mediated interviews (for example, via email or instant messaging) or having to travel to remote locations, which would have posed significant challenges both logistically and in terms of affordability, as my research project has mainly been self-funded.

The initial design of this second stage of the research project took shape around 2008 and the actual fieldwork was carried out between the end of 2010 and the beginning of 2011: the core fieldwork data is therefore situated in time within the 2010-2011 timeframe and its associated technical and normative conditions. Likewise, the mediated discourse analysed to prepare for my fieldwork and throughout the data collection and analysis phases is similarly situated in time. The literature and theoretical frameworks employed in the analysis of the fieldwork materials, however, extend to the present day. Similarly, there are some elements of mediated discourse from more recent times that found their way through to the writing of this dissertation: in specific cases, more recent developments provide additional perspectives on earlier interpretations: when this is the case, I always try to acknowledge and contextualise the different timeframe of more recent discursive elements, and to explain why they are used. Unavoidably, the material conditions associated with such a flexible environment as the Internet change rapidly at times, although appropriation, translation and reassembling tend to operate on longer timescales, as discourses and personal lives do (Slater 2013, p15). In a way, therefore, this is an analysis of the material culture of the Internet for a small set of users and within a limited timeframe, but by engaging in a wider critical discourse my hope is to trace some more persistent lines which, wile drawing on situated stories, extend beyond them and provide elements for a critical analysis of the interplay between people's lives and the ever-expanding scope of the Internet in today's world.

3.2.2 An ethnographic project?

While conducting my initial pilot interviews, I wasn't completely sure about which kinds of insights I could expect to gather from these conversations, nor what exactly to look for: although my project's research question had already been refined through several iterations of critiques and reviews (as well as changing material conditions, such as the aforementioned institutional pressure derailing my initial hope to develop the project as action research), translating that into the practicalities of fieldwork research was still a substantially open process, which I set out to refine through the inevitable — and welcome — productive failures I was expecting to experience during the pilot interviews.

In fact, I wasn't convinced yet about a crucial methodological choice: whether my fieldwork would need to consist mainly of interviews or of a balanced mix of interviews and participant observation. After all, most of the seminal studies most relevant to my project (e.g. Bakardjieva 2005; Miller and Slater 2000) did in fact rely on intensive participant observation, although inevitably also including extensive conversations with their research participants.

Accordingly, when organising the pilot interviews I made sure to ask students to bring along their laptops, if that was their main way of accessing the Internet daily, or to organise our meetings so that we could sit in a quiet space on campus in front of one of the open-access computers provided by the university, so that I could ask students to go about some of their usual tasks on the Internet while I would observe, asking questions if I felt I needed to know more about some specific aspects of their practices and interactions online.

However, as I gained both the necessary confidence to conduct these meetings and a deeper understanding of the boundaries and limits of what I could hope to achieve by way of semi-structured conversations, on one hand, and of observation, on the other, I experienced a substantial disparity between the richness, quality and quantity of insights rapidly emerging from the conversations with students and the relative uneventfulness of their actions in front of a computer.

Granted, ethnography as a full immersion in participants' lives over an extended period of time (Hammersley and Atkinson 2007) cannot be other than a slow, often messy and apparently inconsequential practice, whose results only start appearing over time and by looking at the accumulated knowledge rather than by focusing on every single moment considered in isolation; in digital anthropology as well as other forms of anthropology, participant observation is a long process (Boellstorff, Nardi, et al. 2012, p55). Miller and Sinanan (2014), on the other hand, while discussing the sources of their data about the material culture of webcam, note that in some anthropology circles the interview has been considered secondary to actually participating in informants' daily lives, but acknowledge that in the case of their research interviews played a fundamental role alongside their extensive fieldwork in Trinidad, and further cite Skinner's edited volume (2012) to corroborate their case for the role of interviews, drawing on the volume's contributions examining scope, limits and strengths of the interview in anthropological research, by itself or in concert with other methods.

My methodological reservations, however, were of a substantially gnoseological nature: firstly, my pilot interviewees were often mentioning interesting and relevant facts, practices, ideas or discourses which were nevertheless detached from the immediacy of practices that I could observe directly: for example, the choice to give up a mobile phone-with-SIM-card and to replace it with a SIM-less smartphone used mainly as a VoIP terminal whenever WiFi coverage was available because of much earlier experiences with very high monthly mobile phone charges; secondly, my specific perspective of analysis of life-world Internet includes the examination of both human and non-human agency, which in turn includes both somewhat observable elements such as, for example, affordances provided through web applications as well as less tangible but equally relevant factors such as roles of protocols and economics of the Internet infrastructure: these factors may at times be gleaned through their possible or alleged effects on observed practices, but their appearance would be mostly incidental (for example, through frustrations of the users' experiences which lead to the attempt to circumvent specific issues encountered by exploiting hidden affordances, or visible ones in ways they weren't designed to be used), and participant observation would not be best suited to account for a large part of the non-human agency observed, without substantial questioning of users' accounts and assumptions, which in turn leads back to the need to discuss these through interviews.

Moreover, and somewhat differently from clearly ethnographic studies of Internet users or users of other media in everyday settings (such as Silverstone 1994), my intent was not to investigate users' *practices* but rather *their own accounts and narratives about their practices* related to the Internet, against the normative backdrop of institutional regulations and laws, of economics and of technical configurations. Accordingly, my methods need to take into account the specific object of investigation — the discursive dimension around Internet practices–hence the choice of relying mainly on interviews, even while effectively keeping open the possibility of observing interviewees as they use Internet-connected devices and online apps, although with a qualitatively different aim than a longer-term participant observation in the users' home environment — which in turn, and conversely, in ethnographic studies would normally include a discursive questioning of what the researcher is observing, in a somewhat inverted balance of what I employed throughout my fieldwork.

Therefore, while whether or not the present work can rightly claim to be an ethnography may be open to interpretations, my standing point is that it is rooted in the anthropological tradition and the methods employed were refined through an iterative process to best support my research questions and to allow me to explore in depth the material culture of the Internet in everyday life for a small group of students, by engaging with their own discursive accounts.

3.3 Researching user accounts

3.3.1 Recruitment and selection of fieldwork participants

I conducted three pilot interviews with Goldsmiths students: one while still working on my web application project there, after reshaping my study away from the initial action research plans, and two shortly after the end of my work project; these served as trial interviews to test my intuition to rely on in-depth face to face interviews and to assess how my role as researcher within the in-person setting would influence the interviewees. No actual data and materials from these pilot interviews ended up being used for this study.

The main fieldwork consisted of interviews with fourteen students, conducted between November 2010 and March 2011. All except four students were recruited from a pool of respondents to a call for participants I circulated through the university's departments. Of the four participants recruited directly, two were people I met randomly through life circumstances, and upon learning that they were studying at Goldsmiths I asked them if they would be available to be interviewed for my research; one was actually the wife of a student who had died some months before in an accident; I didn't know him personally but he was part of my Facebook circles and as his Facebook profile was being curated by his wife after his passing I asked her if I could interview her about this and about themselves and their experience of the Internet. The last one of these students is a friend I had known for years, and worked with on several projects: his ways of using Internet-based tools were of great interest to me and I asked him to allow me to capture these through formal interviews. All other students were essentially unknown to me, including the three recruited directly, whose personal circumstances I had not known yet when they agreed to take part in my study and when they were interviewed. A brief profile of each student interviewed is presented in Appendix A of this dissertation.

As outlined earlier, the group of interviewees comprised fourteen students. Relatively small numbers of participants are not uncommon in ethnographic research, as the depth of materials and descriptions that can be gathered through long interviews, direct observation and other ethnographic methods can be considerable even with numerically limited pools of participants. As an example from similar studies, Bakardjieva's (2005) fieldwork included just over twenty participants drawn from the Vancouver metropolitan area, while still providing extremely rich insights about their experience of the Internet in everyday life. Miller's (2011) study of Facebook users in Trinidad presents twelve 'self-contained' portraits, but although the full number of participants is not discussed, interestingly he notes that some portraits are actually composed of details from different individuals, to preserve anonymity but arguably also to group under coherent portraits heterogeneous traits which the author's experience considered to be illustrative of similar distinctive characters.

On a somewhat opposite end of the spectrum, boyd (2014) draws on a much larger sample of American teens, 94 of which were originally interviewed for her doctoral dissertation (2008); furthermore, her articulated data collection methods included reviews of hundreds of online profiles on social networks and overview of mediated discourse online. The core aim of her research, however, was to understand, through the specificities of each teenager's story, a more general transformation of sociality taking shape with the widespread use of social networks in the geographic context of the United States, connecting this with broader social changes.

When operating with qualitative data, the size of the group of research participants is much more related to the aims of the research project and to methods employed throughout the fieldwork than it is when dealing with quantitative data, whose sampling logics are radically different and often need to rely on large numbers of observations in order to provide statistically robust models of the behaviours analysed.

Whereas each participant is, strictly speaking, accounting only for themselves, the extent to which the research aims to identify overarching trends, propose generalizable abstractions, or pursue a minutiose analysis of particularities in order to gain a deep understanding of each individual's unique motivations, suggests different logics when establishing an optimal number of research participants.

A case study, intended as a situated analysis of a small subset of a larger group or phenomenon, can uncover valuable insights which will still require validation through a broader analysis if their general relevance to the full population is enquired; conversely, a broad cohort of fieldwork participants does not necessarily make it possible to understand, for example, the deeper motivation of each individual if the observation or questioning don't probe specifically for these aspects.

In the specific case of my research, the aim of understanding users' practices and motivations through their unique accounts, as well as engaging these accounts in a double critique (contrasting them with technical constraints and broader issues of power and consumption economics, as well as negating assumptions incorporated in technology design with users' reactions to constraints and lack of information) led to the specific choice of number of participants, context and research methods: as highlighted by Brannen (1995), all these concur together in determining how robust the findings can be judged in the case of qualitative studies, where issues of generalizability (as understood in quantitative research) are replaced by different concerns:

> In the instance of case studies for example, the issue needs to be couched in terms of how far the findings can be extrapolated to the theory that the research has been designed to test. (Brannen 1995, p9)

Furthermore, when one of the aims of the theory is to actually attempt to negate assumptions that are commonly accepted or to disassemble structures that have become black-boxed, a single case study example that negates a widespread assumption can invalidate it, or at least question it by forcing the researcher to elucidate the specific conditions under which the assumption can still stand while being negated in practice in other cases.

3.3.2 Natives of Read/Write Internet? Demographic considerations

As outlined in the previous sections, of all the fieldwork participants only one was well known to me (as a friend) — all the others were essentially unknown to me and I didn't search for information about them before our interview meetings, in order to approach the interview setting with no prior, even cursory, knowledge of traits of their life stories, interests and occupations.

Of the students selected between those who responded to my call for participants, I only knew in advance for which kind of degree they were studying (undergraduate, postgraduate certificate in education — PGCE, Masters or PhD) and in which department, and their age range: whether they were 25 or younger, or, if older, their age within ranges of ten years each (25-34 and so on); this information was gathered through a basic online questionnaire which students responding to my call for participants were invited to complete. Unless for specific reasons related to elements of their interviews that needed clarification, I didn't probe for more specific demographic information during the interviews, letting students disclose what they thought was of interest in relation to the accounts discussed.

20% of participants were BA students, 60% Masters students, 6.5% PGCE students and 13.5% PhD students. 13.5% of participants were between 35 and 44 years old, 33% were younger than 25, all the rest (53.5%) were between 25 and 34 years old. Given the number of participants and the character of my research, the exact age of each participant was much less interesting than their life stage, as well as assessing the extent of their exposure to pre-Read/Write Internet.

Students who were 25 or younger in 2011 would have been 14 or younger in 2000, which would have limited any significant exposure to pre-Read/Write Internet, at a time when households would connect to the Internet mainly through a family desktop or laptop and kids wouldn't typically have portable devices they could use in their own rooms, thereby being able to experience the Internet independently from an earlier age: roughly a third of the participants were therefore 'native' of the Read/Write Internet. As discussed in the historical overview of Read/Write Internet in Chapter 4, the kind of interactivity and two-way flow of information available through early Read/Write Internet apps was obviously very different from what my fieldwork participants were accustomed to in 2011, but nevertheless the idea that the Internet was a Read/Write technology was present since the beginning of these students' experience of the Internet. The remaining two thirds of the participants could have had some degree of exposure to the pre-2000 Internet (some of them reported that they only started using the Internet more recently), although very few references were made to experiences of pre-Read/Write Internet, thereby limiting the ability to assess how the shift towards a Read/Write Internet had changed participants' attitudes; an historical understanding of this kind is nevertheless beyond the scope of the present work and the few references in this sense that were made just provided more context to contemporary practices of the participants who discussed them.

3.3.3 The ordinary user

A discussion of the process of selecting research participants also needs to include the qualitative composition of the group, beyond its numeric extent. Again, qualitative and quantitative research rely on distinct logics and needs (Brannen 1995, ibid.): the former can meaningfully and profitably employ strategies of purposeful selection (which Brannen refers to as theoretical sampling), rather than purely statistical sampling: that is, a targeted choice of cases — or rather, participants — aimed at maximising the prospects of obtaining accounts that are rich and relevant (as measured against the aims of the research and within its specific context).

In the case of my research, the initial difficulty I experienced in trying to recruit enough participants was addressed through a broad circulation of my call for participants through all the university's departments, as discussed in the previous section, which led to the selection of a small group of participants within the specific population of the student cohorts accepted at Goldsmiths during the year of my fieldwork. As outlined in the previous section, the only two interviewees approached directly because of previous knowledge of some of their Internet-related practices were chosen through targeted selection. Another attempt at this — stemming from my interest to personally revisit at a decade's distance Miller and Slater's (2000) seminal research on Trinidadian users — did not have success: through my network of personal acquaintances originating from my initial action-research project attempt, I asked a lecturer in Caribbean studies with whom I had occasionally worked years before whether she would be able to put me in touch with Trinidadian students for the purpose of my fieldwork, but she wasn't able to do so because no such students were enrolled in study programmes at Goldsmiths during my fieldwork year.

Of the many axes alongside which it is possible to unfold targeted selection of participants (e.g. gender, age, cultural background...), the one which raises issues most relevant to my project is that of the technical competence of users.

Bakardjieva (2005), for example, building on de Certeau (1984), describes as the 'main character' of her work

> [t]he 'ordinary man' and woman who is not involved as a professional (engineer, programmer, designer, etc.) or decision-maker in the industrial, commercial or service sectors developing computer-networking technology. (Bakardjieva 2005, p9)

The extent to which research participants are professionally competent in the topics investigated, either as practitioners or as researchers themselves, must be a core concern when designing ethnographic research: if 'insiders' may provide richer accounts than non-specialist users, there is a concrete risk that they may introduce significant bias in findings, even when perfectly well-intentioned and even when taking into account that different professionals in the same field may have very different opinions and may enact their own role in the field according to widely different practices.

Including specialists as research participants or informants is not an issue per se: in fact, they may be selected specifically *because* of their role, insider knowledge and competence in studies that focus around experts: Coleman 2012a is an example within a field close to the present study, but explicitly concerned with hackers and their practices rather than with ordinary users.

Moreover, the role of specialists/experts poses more complex and nuanced issues in recent (post-2010) Internet studies than it does in most other fields of research related to media and communication technologies, simply because of the relatively high proportion of the general population who has been introduced at some point of their formal or informal education to at least simple programming scenarios, or had a chance to learn some elements of computer programming out of their own interest, as the widespread availability of personal computers and online free learning resources help to lower the barrier to entry, making it possible to gain some basic experience with minimal investment in terms of resources and time. Even when dealing with relatively small groups of users, Internet researchers are much more likely to come across users with some degree of technical expertise, compared to researchers investigating other media such as radio, television, or mobile phones, whose underlying professional fields are much less accessible to the wider public (in the case of the UK, for example, as of December 2013, 54,000 people were employed or self-employed in the economic sector 'programming and broadcasting activities', as opposed to 623,000 — or almost twelve times as much — in the 'computer programming, consultancy and related services' sector, according to Office for National Statistics (UK) 2014a and Office for National Statistics (UK) 2014b).

In my own research, my initial aim was to interview non-expert users, as I wanted to understand how the ingenuity of users without deep understanding of the technologies surrounding them would lead them to performatively interpret affordances, mediating them through representations coming from prior experience and mainstream discourse, working with technologies that could be seen as 'black-boxed' but still operating around and through them to reassemble their very own experience of the Internet. Interviewing expert users would have risked to operate a partial foreclosure of the users' representations of ways to reconfigure their Internet environment, leading to reaffirmations of known (from my own experience as web developer and from my exploration of the mainstream discourse as part of the present work) representations considered as 'best practices'.

In practice, I didn't have to really deal with this issue as none of the students who responded to my call for participants belonged to any of the categories listed by Bakardjieva (above) nor had an extensive professional knowledge of the topic of my research by way of being a researcher in the same field: however when exchanging introductory emails with the students who ended up being interviewed my intention, in accordance with my research design, was to exclude from my fieldwork students who were clearly professionally trained as software developers, engineers or designers. One of the students in the pilot study and one in the main fieldwork cohort had worked as developers, but in their own account this was a tangential career moment rather than expertise they had specifically trained for.

Throughout my fieldwork, a more nuanced and situated view of the 'ordinary user' emerged: whereas the theoretical references are discussed in the literature review chapter and the substantive details are analysed in a later chapter, it is nevertheless useful to give here a brief overview because of its relevance to research methods.

Besides the two students mentioned above, who had worked as web developers at some point earlier in their lives, around one third of the other students I interviewed discussed some degree of involvement in software development practices that went beyond the mere reconfiguration of apps or Internet-connected devices within the boundary of options kept open by the developers: either by coding some simple data analysis and visualization environments, by learning how to operate on the command line of GNU/Linux virtual machines to install, configure and manage some obscure Internet services, or by building custom GNU/Linux distributions tailored to the use within workshops focused on Internet-based distributed video making, they had invested a conspicuous amount of their time to peruse online resources, to ask for advice online or enlisting friends to help them gain some basic proficiency in some relatively advanced programming or systems administration tasks linked to their interests.

Although they would arguably not be considered 'expert users' as their experience was limited to circumscribed contexts and not part of formal education or training nor aimed at pursuing a career in the field of software development, their involvement in expert practices highlights two axes alongside which the profiles of ordinary users can be further problematized whilst dealing with Internet studies: firstly — as in other fields of knowledge and practice — researchers can argue for several ways in which to differentiate experts from non experts (e.g. by way of formal education or considering whether their practice is part of their daily job rather than being an occasional activity or a hobby), though when dealing with materials emerging from an ethnographic investigation and taking into account motivations, justifications and users' attempt at making sense of their own actions, expertise can also be seen as a continuum, with some non-professional users being able to gather significant knowledge and expertise relatively quickly, allowing them to operate proficiently within specific contexts whilst not being considered professionals in the field. Secondly, even professionals within Internet professions are now expected to gain a much deeper understanding of their specific sub-field than in earlier years, in order to be competitive in an increasingly specialized job market: consequently, their expertise in other sub-fields may be very limited and not dissimilar from that of users who are not professionals in any Internet-related field.

Methodologically, these insights translate to the necessary awareness that even when dealing with 'ordinary users', there can often be at least a latent competence and awareness of some subtle mechanics operating behind what may be received as a black-boxed artifact by ordinary users; specific care must then be exercised while selecting participants or while interviewing them, in order to make as explicit as possible latent traits of competence that could potentially introduce expert bias even when users are apparently 'ordinary users'.

3.3.4 The interview process

I sat with each student over one or two interviews of one our each. All the interviews were held on campus, inside the local library or at the campus café, except for the two interviews with my friend, who was interviewed first in his home setting and then on campus at a teaching room after a lecture. I judged that the neutral setting of the university campus would be an acceptable compromise between my desire to recruit interviewees quickly without having to overcome possible reservations about letting me enter their personal home environments and the ability, on the other hand, to gain a deep understanding of their accounts. Most of the interviews were conducted in front of a computer (either a library computer or the student's own laptop) so that I could ask the student to show me something specific they were referring to throughout the conversation.

My goal was to conduct two interviews with each student, normally about one week apart in order for myself to have time to review the recording of the interview and identify issues I wanted to probe more in depth in the second interview; this short interval between the interviews also ended up leading some students to spontaneously add more details or clarifications to what was discussed during the first interview, having had some time to reflect over it. Due to students' schedules, in three cases we weren't able to meet again after the first of the two planned interviews. In all these three cases, however, the depth of discussion throughout the first interview was quite considerable and I was extremely satisfied with the outcome of the interviews and the wealth of material they provided to my study.

Conducting each fieldwork interview, as I expected and looked forward to, was a very intense and rewarding process in itself, even before looking back at the contents discussed, listening to the recordings and starting to critically analyse the interview data: careful planning and preparation was therefore vital in maximising each student's participation by keeping the interview running smoothly and focused, while leaving space for participants to bring the conversation towards relevant topics that I could not have anticipated before knowing about at least some key traits of their personal experience and history.

In designing the interview structure — and in refining it through the pilot interviews — I combined useful elements from relevant interview techniques. The basic structure used was that of the long, semi-structured interview, with a simple and open–ended questionnaire which I developed mainly through successive iterations during the pilot study. I found McCracken's (1988, p25) recommendations for the use of an interview questionnaire in qualitative research (ensuring a standard ordering of topics, 'care [...] of the prompts necessary to manufacture distance', forming a scaffolding for 'direction and scope of discourse', allowing the interviewer to maximise their focus on the interview situation avoiding distractions) relevant especially in the case of in-person, real-time interviews, where the urgency of every moment and the complex interactions between the verbal dimension and body language create a considerable pressure on both interviewer and interviewee. However, as my purpose was mainly exploratory (trying to let the participants freely talk about the material conditions of their Internet experience) rather than focused on specific aspects such as attitudes towards a narrow set of phenomena, the questionnaire was necessarily only serving as a rough guidance, to make sure I would cover all the relevant basic terrain and could devote all my attention to each student's accounts, while in practice leaving the scope of each question and spontaneous or prompted follow-up as open as possible, within the limits of reasonable relevance.

Although my core experience prior to this research project was mainly with long (2–3 hours on average) open–ended life story interviews aimed at exploring participants' attitudes and values (Marradi 2005), through the pilot phase of my fieldwork I realised how useful it was for me to build a simple interview structure which served both as guidance (and, initially, reassurance while still building up confidence) and as a way to elicit often passionate accounts from students. The situated context of an university campus as backdrop of each student's current life stage constituted an important discursive frame: accordingly, basic elements of the questionnaire served the purpose of exploring each student's motivation for joining their specific course and of starting to explore their daily routines — if any — related to the use of the Internet.

On the other hand, my previous experience with long life story interviews, corroborated by the pilot study, convinced me that some circumscribed elements of life story interview methods (Atkinson 1998) would still be highly relevant and, accordingly, I made sure that when students would feel comfortable with this, I could gently prompt them towards narrating personal stories (most often from their recent past) which could help better framing their accounts more strictly focused on their experiences of the Internet. My intent here was not so much to try to trace motivations and intentions back to important experiences distant over time (although this happened in a few cases by spontaneous, unprompted associations made by the interviewees), but rather to trace each person's experience through the material conditions of their everyday life, which often included objects, ideas and events apparently disconnected from the Internet but which were then giving a richer context to decisions and practices relevant to my object of enquiry. This included past family events, personal life turning points, work experiences and other foundational moments (ibid., pp22-25) whose role (and existence) I could not have foreseen as the participants were substantially unknown to me. Additionally, I expected that a controlled freedom to uncover personal stories with a high emotional load would make interviewees feel that their experience was highly valued by myself as a researcher, and consequently that they would feel welcome to actively explore latent connections rather than simply responding to interview questions in a more clinical way.

In this respect, I tried to fruitfully combine the basic methodological rigour of a questionnaire-based interview with the serendipitous fruitfulness of open–ended and life story–based methods, with the important distinction that the 'life' I wished to explore included both human and non–human elements, whose voice I hoped to be able to overhear (Kohn 2013) through the interviewees' passionate accounts.

Throughout the discussion of user practices (Chapter 6) based on this fieldwork, details and names have been changed to preserve the participants' anonymity. To anonymise participants' names while retaining a reference to their geographic and cultural connections, I chose names by consulting country–specific baby names websites, according to the country of origin of each student.

3.4 Making sense of fieldwork stories

3.4.1 Prior knowledge and the role of the researcher

However rich and enlightening the data gathered through fieldwork may be in shedding light over individual, situated stories, one of the tasks that the researcher must face is trying to make sense of the complex network of interrelations between participants' accounts, normative frameworks, existing knowledge and interpretations, and theories the researcher may want to test, critique, validate or further extend and corroborate.

My work is in fact an attempt to interpret a multitude of discursive accounts, by way of questioning their internal structure, coherence and logic, of analyzing their cumulative (and necessarily comparative) knowledge, and of contrasting them with existing accounts. This process is unavoidably marked by the role of the researcher, whose previous knowledge and experience may at the same time introduce biases (which need to be acknowledged and honestly taken into account) as well as provide an unique and hopefully productive environment for the analysis of these multiple sources and layers of data and interpretations.

In my case, my relevant background is a professional experience in various roles as Information Technology freelancer for over a decade at the time of my fieldwork: mainly as a designer and administrator of web services infrastructures at first, and more recently as a full-stack web developer; my academic background, instead, is in the fields of philosophy and the social sciences.

The actual contrast between what I had been reflecting on through my academic upbringing and experiencing through my professional practices, on one hand, and the way that web services and the economic forces behind them started influencing the lives of larger portions of Internet- connected users since around the mid-2000s, on the other, provided the initial motivation for my decision to start this research project.

My analysis of the fieldwork data, therefore, can be seen as both being informed by and situated within my prior knowledge of the technical and economic structure behind the Internet, although my responsibility as a researcher is obviously to strive to maintain an approach as critical as possible.

Furthermore, besides my practical knowledge of the technologies underpinning communications over the Internet, I am a user of the Internet in my daily life just like my research participants, which again both gives me a situated perspective over the phenomena I investigated throughout my fieldwork but can also introduce further assumptions that need to be made explicit and acknowledged.

And finally, as part of my research and even before starting the fieldwork proper, I explored and analysed a large quantity of discursive formations that engaged both in describing, analysing, explaining and shaping the technologies and practices that I focused on while conducting the fieldwork interviews.

All these — professional, personal and research experiences — could be considered as sources of prior knowledge. They were useful to me in order to pursue an initial tentative design of my research, through the understanding of the existing debate and the different points of view within it. The issue of whether a researcher should approach the field with as little prior knowledge as possible of it — so to avoid biases — or whether at least a general understanding of the field and the issues that may be encountered is desirable is an ongoing debate within the social sciences. Hine (2000) highlights how prior experience can help researchers in the field of Internet studies conceptualise the general structure, power struggles and main issues that they may be encountering throughout their fieldwork. The researcher's existing knowledge can therefore be useful insofar as it helps forming a rough understanding of the participants' situations and to establish initial rapport, upon which the actual participant observation or interview process can be developed, rather than trying to anticipate every possible issue that may be experienced while doing fieldwork.

Besides strictly virtual ethnography, in the case of ethnographic research of Internet users conducted in person rather than online, prior knowledge of the environment in which users are situated can help in anticipating key trends and even possible ways in which interviews may risk being steered towards unproductive discussions of common knowledge with little personal input from the interviewees. Throughout my fieldwork, the awareness of issues encountered during the initial attempt at developing my project as action research (as outlined briefly at the beginning of this chapter), both in terms of knowledge emerged from questionnaires submitted by users and in terms of assumptions regarding the use of Web 2.0 tools and their integration through the web application I was developing, constituted a precious body of knowledge, at least about what to (try to) avoid while conducting the core of my fieldwork at a later stage.

Moreover, prior to my role as web developer at Goldsmiths, a very brief work engagement at the same university as a night shift IT help desk assistant unexpectedly proved to be precious training for my ability to empathise with users and their daily frustrations with Internet-connected devices in general: although I had been providing support to users from remote as a freelancer for several years, only when I worked in close contact with students as a support assistant often on call during the night, when most other source of help would have been unavailable, I managed to get a deeper understanding not only of the issues they were experiencing, which were often banal and uninteresting per se, but mostly of the impact that malfunctions and related frustrations had on their material everyday life: this turned out to be a very useful asset while conducting interviews, by helping me to probe with questions beyond the immediate account of facts, to try to understand users' motivations and deeper feelings about what they were discussing with me.

3.4.2 Between autoethnography and participant observation

Whereas in the previous section I have focused on the role of the researcher with reference to knowledge (and prior knowledge), there are other important ways in which a researcher can be connected to various extent to their fieldwork environment and participants, which need further discussion.

In social research, a lab–like setting is much less common than in the natural sciences and in computing and engineering disciplines: when interacting with participants, a reseacher is almost invariably disrupting to some extent the field, for example through non–verbal communication, through questions and demeanor, and ultimately through their bare presence, even when this is that of a silent observer. These elements may influence researcher, participants and their interactions; other factors, however, may contribute to constituting more nuanced relationship with the individuals or groups being researched and with their environments and cultures, making to various extent the researcher and their voice a substantive part of the ethnographic enquiry: deeper cultural, emotional, professional connections; whether participants and informants interact with and perceive the researcher in roles other than their primary one of researcher; communicative traces that a researcher may articulate within the field itself while conducting fieldwork beyond their interactions.

Throughout the three distinct but interconnected stages of my own fieldwork, my involvement with field and participants was configured in ways that extended to my role as professional software engineer, as participant in projects developing alternative rationalizations and reflections upon these (presentations at technical conferences, coordination of workshops), and as a member of the recursive public of hackers whose constitution I explore in the final two empirical chapters of this dissertation: my role must therefore be considered, and analytically deconstructed, as substantially different from that of a researcher culturally, professionally and emotionally detached from his object of study.

An useful framework of reference to articulate such relationships can be identified in the methods linked to autoethnography: although definitions of autoethnographic processes, methods and theories are widely conjugated by different scholars (a review that problematizes the uniqueness of approaches can be found in Denzin 2006), a clear overview is proposed by Anderson (2006):

> Put most simply, *analytic autoethnography* refers to ethnographic work in which the researcher is (1) a full member in the research group or setting, (2) visible as such a member in the researcher's published texts, and (3) committed to an analytic research agenda focused on improving theoretical understandings of broader social phenomena.

All these traits can rightfully be considered to apply to my fieldwork: (1) I was a member of the 3D Graduate project (Chapter 5) as project developer and as a free software developer I have been a part of the recursive hacker public whose discussions and projects are discussed in Chapters 7 and 8[4]; (2) in a literal sense, the present work makes my membership of these settings and groups visible, but in a more complex sense, also the output of my FLOSS software development activities (software code, documentation, code review, user support) and my own reflections upon these (presentations at technical conferences) are making my belonging to the researched group(s) visible, and were/are available to the other members; (3) my political commitment to user agency in lifeworld Internet and to decentralized alternative rationalizations of mainstream Internet affordances is informed by my analytical work developed through my research.

In practice, therefore, autoethnography does not *constitute a choice* of methods as much as it *follows* my commitment to operating as a hacker/maker within the domain that I chose to investigate: the research focus itself follows my practice–based interests and the realization that the domain of lifeworld Internet is currently less explored than the other intersections of computation and the social that I outlined in the previous chapter. Instances of recent autoethnographic research highlight how autoethnography is often similarly framed within predicaments and situations that follow tragic life events (Weaver-Hightower 2012) or life choices (Stanley 2015), constituting a somewhat obligatory choice that shapes the researcher's focus and methods. At the same time, however, Anderson 2006; Atkinson 2006; Denzin 2014 highlight the importance of employing an analytic (rather than 'evocative') approach in autoethnography, in order to ensure analytical validity and methodological soundness of research findings (Anderson 2006, p387).

[4]It must be noted, however, that the distributed, large–scale and heterogeneous nature of the new recursive public of hackers of which I feel part naturally complicates the identification of stable group memberships and alliances.

A strategy that proved fruitful throughout my research consisted in re-analyzing the fieldwork materials originally collected at some time distance, applying retrospectively the interpretive framework of the computational turn of the Internet. In the case of the 3D Graduate project, this allowed me to situate my role of developer committed to user–focused interpretations of the Read/Write character of Web 2.0 within the broader context of transformations of computational agency that became clearer only five years after the start of my project. In the case of the analysis of alternative rationalizations and decentralized projects of the final two empirical chapters, my engagement with discursive materials (outlined in Appendix B) was complemented by the conceptualization—which I developed while still undertaking the fieldwork but after I could reflect back on the materials gathered in the first couple of years of it—of the recursive nature of the hacker public of which I am part: this conceptualization is articulated in the present work through the analysis of the technical materiality through which this recursive public interprets and reproduces itself (Chapter 7). Whereas the actual methods used throughout my fieldwork were in practice not dissimilar from those of participant observation, it is the reflective and analytical framework informed by my own direct engagement with the technical materiality of computation that constitutes the more direct contribution of autoethnographic awareness to my understanding of the phenomena studied.

Lastly, direct engagement with the field in capacities other than the primary one as researcher is also a trait of the domain of 'aca-fandom' (Hills 2002; Jenkins 1992); I contend that this is not, however, the setting of my own research: although on one hand I am clearly committed to the values of free software and of alternative rationalizations of lifeworld Internet, the recursive character of the hacker public of which I am part constitutes a decisive distinction from the domains of fandom: whereas fans typically engage with *cultural artifacts* through discussion, appropriation and remixing (processes that are themselves transformed by the availability of cultural artifacts in digital formats and of computational tools to procure and manipulate them: Jenkins 2008; Jenkins, Ford, et al. 2013), the recursive public of hackers examined here is fundamentally operating by reshaping the very infrastructure that enables its existence and cultural and social reproduction, through political and technical contestation rather than mainly through cultural appropriation. These distinctions will be further deconstructed in Chapter 7 through the analysis of the constitution of the new recursive public of hackers whose motivations are then traced through the projects and challenges analyzed in Chapter 8.

3.4.3 Beyond the field: following the actors

Although the core of my research data is constituted by the analysis of my own development project (and related records of email exchanges and other written documents) and by the accounts of the students interviewed throughout my fieldwork, I quickly realised — already at the fieldwork stage and even more compellingly while analysing the research data — that in order to fully explore my central research question I needed to contextualise the participants' accounts by somehow 'following the actors' (Latour 2005, p12), specifically the human and non-human ones involved in the assembling of lifeworld Internet in a context much broader than my fieldwork; this section presents an overview of the research data I used for the analysis of this broader context, the motivations for its use, the limits of this data, and how fieldwork and non-fieldwork materials are related analytically. A detailed overview of the specific groups of source materials used is presented in Appendix B.

Whilst my research participants were re-assembling their individual lifeworld Internets, they were far both from starting from a *tabula rasa* and from using fully shaped and coalesced technologies: instead, they were appropriating, reinterpreting, contextualising and reshaping affordances and interpretations that had been built by innumerable and heterogeneous actors through a fast-paced but nevertheless progressive evolution across over a decade: to consider the materiality of the Internet as re-assembled by each user therefore inevitably means having to trace how the intentions and actions of the networks of actors *behind* the web applications available to users had been inscribed in the technologies themselves.

In other words, whereas a large part of previous literature dedicated to the social aspects of Web 2.0 technologies (as outlined in Chapter 2) regards these as substantially commoditised and black-boxed affordances that users then appropriate and re-interpret, therefore privileging the analysis of agency of end-users against the backdrop of the applications available to them, my argument is that in order to comprehensively trace powers and agency involved in the re-assembling acts of users one needs to follow the actors *past the ethnographic accounts of users* and consider how the technologies that permeate their daily lives are themselves constantly being reshaped at different levels and by multiple actors.

As such, my research aims to follow the intimate details of how lifeworld Internet has been taking shape by engaging (in Chapter 4) in a brief historical analysis of its evolution, with particular attention to the technical aspects and to mainstream discourses about the use of the Internet in everyday life, and by including (in Chapters 7 and 8) the reflective discourses of the people (developers, designers) who ultimately build these technologies either as part of their daily jobs or as part of informal collaborations and side-projects with other developers.

For the analysis of the reconfiguration of computational agency through the computational turn of mainstream Internet in Chapter 4 my sources are the relevant academic literature as well as expert accounts on web publications (blogs and magazines) and throughout discussion forums; throughout the recent evolution of Read/Write Internet issues such as practical aspects of software engineering, marketing and ”startup culture” have had a much more profound and immediate impact than scientific innovations; moreover, the time and effort needed for researching and producing academic histories of software also implies that the materials available on very recent developments is usually scarce: therefore my research also includes non-academic, expert accounts, whose validity I strive to ensure through my own technical expertise in the field and through combined examination of distinct accounts on the same subject. Using this combination of sources, an overview of the transition of mainstream Internet infrastructure to configurations much more reliant on computation than at any previous stage is in my view necessary also to contextualise more precisely the accounts and claims of fieldwork participants.

Moreover, besides the important differences *within individual web applications across time*, differences *within individual genres of web applications* are also a factor of distinction that is often overlooked; an ethnographic study of the re-assembling processes operated by users cannot rest on the simplistic view that, for example, ’blogs’ is a substantially uniform genre of web applications: users who incorporate reading or writing blogs in their everyday Internet experience do so in very different ways not only according to their individual circumstances, interests and skills but also according to the ways in which different blogging platforms expose their functionality in unique ways that may or may not fit users' expectations and needs. My technical overview in Chapter 4 is meant to also address this common shortcoming in order to better inform the ethnographic analysis of user agency.

For the analysis of reflective discourses of hacker involved in the development of the alternative rationalizations of lifeworld Internet developed in Chapters 7 and 8, my sources are mainly the discourses of web developers, designers and software engineers as emerging from online discussions and from formal and less formal print and web magazines written by and for these Internet professionals themselves, with the specific non-technical focus of trying to develop a shared understanding of the role that web professionals have in today's world, and which relies considerably on the Internet as a conduit of information in the most disparate contexts. Although the inclusion of these discourses may partially sound like a posthumous attempt to turn the inexorable procrastination efforts of a graduate student who spent uncountable hours absorbed in such publications while supposed to be working on his dissertation into structured and productive procrastination (Perry 2012b), in fact my immersion in such publications, initially only aimed at improving my own understanding of my role as a web developer thanks to the accounts of other fellow web professionals, actually enabled a pivotal shift in the way I developed a broader understanding of the complex networks behind the Internet affordances that permeate users' daily lives. In hindsight, my exposure to the reflective accounts of hackers is what most contributed to my realization that even when focusing on the common user, tracing any process of reassembling would provide only a partial view unless the 'sweat and tears' involved in the production of software, and the often mundane materiality of developers' own daily lives and motivations, are also analysed.

As the production of web and mobile applications became an endeavour attracting fast growing numbers of developers, starting with the spread of Web 2.0 applications but especially after the iPhone's launch in 2007, web professionals showed an increasing interest in trying to define their roles within new professions as well as within a globalized world in which the Internet plays a fundamental role: several formal and informal publications devoted to the self-reflective accounts of these professionals have been contributing to the shaping of this consciousness, presenting stories of individual developers and designers as well as their ideas about what it means to them to be involved in the creation of software which is used sometimes by millions of users in everyday life. Compared with the vast array of technical publications that for a much longer time have contributed to debating and circulating best practices in software development, these more recent publications often focus on the mundane histories of individuals, tracing how life events and personal convictions contribute to their choices of projects to work on and how they shape the very ways in which they design and develop software that underpins the Read/Write Internet.

Whereas earlier developers of desktop (as opposed to web) applications would typically be working on software that only in some cases would be used very broadly and also by the developers themselves, web professionals are themselves very often users of a wide array of web applications in their daily life; as they are exposed to similar frustrations and joys of ordinary users, and — more importantly — to similar external and internal factors contributing to the reconfiguration of their computational agency in everyday life, exploring the links between developer discourses and user experiences represents a relatively novel research opportunity: specifically, this partial overlap of roles (developers and users) can inform the analysis of the democratic potential of user engagement in the production of software within the perspective of critical theory of technology (as outlined in the final section of Chapter 2).

Any use of research data from hacker discourses related to Read/Write Internet, however, needs to take into account several limitations. Firstly, these accounts largely reflect the severe gender bias of the broader computer science field: although female participation both in technical and non-technical discussions is slowly increasing, also thanks to some of the non-technical publications' efforts to promote female developers as role models, male developers still constitute the vast majority of contributors to these publications. As Brach, the editor of one such publication, recently commented while addressing his attempts to overcome gender bias in his magazine:

> I'm not sure what more I can do than intentionally including more women

> on my 'potential contributors' list. It doesn't seem to convert into a more balanced line-up though. [...] And that's where I am today. After (soon) seven issues, I'm not one bit closer to making Offscreen a more gender-balanced publication, and I don't know what else I can do. (Brach 2014)

Other limitations similarly stem from the fact that the population of Read/Write Internet developers is still predominantly white and middle class. Although a wide array of personal beliefs is represented within these geek discourses, including very liberal views that promote inclusivity and respect for user freedoms, nevertheless first-hand accounts of developers from the global South and from poorer backgrounds are practically absent.

Finally, a further limitation lies in the fact that English is the vastly dominant language both in technical forums and in non-technical discussions: the voices of developers who are not proficient English speakers are, accordingly, underrepresented. This issue is partly mitigated by the fact that some of my sources are edited publications, which can accommodate interviews to non-native speakers who otherwise would not feel comfortable with expressing their thoughts in English through their own blog posts.

When embracing an ANT perspective, the fieldwork and non-fieldwork sources of data outlined above could be considered distinct facets of a single, interconnected corpus, with different actors contributing interpretations and acts of reassembling within their own field of action. In practice, however, this cannot translate to a unified framework for the management and analysis of the research data: although the Read/Write Internet applications used by the research participants are the nexus of the network where the accounts of different class of actors meet, there is little real overlap between these distinct sets of data. Most of my research participants were not familiar and often not aware altogether of the reflective accounts of developers behind the applications they were using, and likewise web professionals normally enter in contact with user accounts only through user experience workshops conducted as part of user research, but do not engage in constant dialogue with users. Accordingly, throughout my work I maintain an analytical distinction between the different sources of data: fieldwork materials underpin the analysis developed in Chapters 5 and 6 according to the core analytical focus of my dissertation; public discourses of hackers provide the contextual background for the analysis of Chapter 4 and the core material for the exploration, in Chapters 7 and 8, of computational agency through the work of hackers involved in the development of alternative rationalizations.

Chapter 4

Mainstream *lifeworld Internet*: material architecture and the role of computation

4.1 Introduction

This chapter examines the materiality of mainstream Internet, with a specific focus on the domain of lifeworld Internet: the sphere of practices and technical affordances that involve the use of the Internet throughout the everyday lives of common users.

The material approach to this enquiry is aimed at understanding how the configurations that attained widespread adoption in recent years have been shaped by design choices, agendas, narratives and software engineering practices, and how these configurations, conversely, sustain specific user practices while—implicitly or explicitly—discouraging or outright preventing others.

Two central aspects of this analysis will be developed: firstly, the material architecture of lifeworld Internet is examined, highlighting connections to the early Web 2.0 narrative and its successive developments, as well as to the historical process through which a client/server architecture—rather than a peer–to–peer one—prevailed across mainstream

Internet, sustaining specific kinds of practices and of agency. Secondly, the analytical framework of *computational turn of the Internet* is introduced, arguing that across the timespan of my empirical research a decisive shift happened, making the Internet in 2015 much more vitally reliant on computation than it was in 2005 (or ever before); whereas in this chapter I focus on how the traits of this computational turn were developed and seized by capitalist agendas, in the following chapters I will analyze how these traits *also* enable new ways of reassembling alternative rationalizations.

The aim of the chapter is therefore to develop the conceptual framework needed to deconstruct the transition between the early promises and expectations of end–user choice and control woven into the Web 2.0 narrative and the actual consolidation of computational agency under the control of large Internet and advertising corporations; this transition is then analyzed in detail in Chapters 5 and 6 through the core fieldwork materials. The specific focus of this analysis is not on the *consequences* of this reshaping of computational agency but rather on its *material foundations* and on how these can be analytically framed; this approach is then further developed in Chapters 7 and 8, in which I discuss how hackers involved in alternative rationalizations have been able to interpret and exploit the changing computational context while seeking to develop affordances and configurations aimed at liberating lifeworld Internet from the domination of corporate and government agendas.

The central argument is that the transition from Web 2.0 promises of openness and choice to the current mainstream configurations as well as the potential for development of alternative rationalizations both revolve around the role of computation as the site of mediation between the social and the technical: through it, values and norms of relevant social groups are negotiated and inscribed in technical form, configuring spaces for computational agency that are historically and materially determined and contested. A distinctive trait of this approach is that computational agency is here considered not only in terms of capacity to act enabled (or restrained) by the computational layer, whether pertaining to human or machinic agents, but also as the recursive capacity to reconfigure the computational layer itself, therefore in turn affecting one's own and others' computational agency.

This chapter is composed of four sections. In the *first section* I trace how the architecture of lifeworld Internet is sustained by Read/Write affordances, in turn rooted in the early Web 2.0 narrative of 'network as a platform'; whereas this has widely been interpreted (and developed) as a platform for human participation, I argue that an interpretation that includes both human and non–human actors as equal participants in complex interactions mediated by computation allows to position the widespread practices of centralization, proprietarization and surveillance of Internet affordances as being deeply interwoven with the practices of end users. In the *second section* I build upon this articulation of human and non–human actants in order to analyze how the telos of progress that permeates Web 2.0 discourses, visibly promoted through the rhetorical artifice of version numbers (2.0, etc.) describing successive evolutionary stages of technological form, contributed to formulating a one–dimensional narrative that privileges the idea of linear evolution and by doing so colonized mainstream lifeworld Internet, while relegating to subaltern position alternative models of computational agency such as those discussed in Chapters 7 and 8. In the *third section* I analyze the tensions between two different broad configurations of communication, information exchange and computation enabled by the low–level protocols of the Internet—a client/server architecture and a peer–to–peer architecture—as well as the historical processes through which the client/server architecture became the largely dominant configuration, in turn determining a range of ways in which individuals could appropriate the Internet as a two–way (Read/Write) medium within the domain of everyday life. In the *fourth section* I introduce the key traits of the computational turn of the Internet as can be observed in recent (post–2010) discourses and practices of web and Internet software engineers, and I develop the analytical framework of computation as site of mediation between the social and the technical, which is used throughout the following chapters.

4.2 From Web 2.0 to Read/Write: the architecture of lifeworld Internet

As discussed in Chapter 2, the year 2000 is conventionally considered a turning point in the establishment of scholarly and mainstream discourses that acknowledge the tight integration of the Internet in everyday life that was increasingly visible in user practices at that time, in a transition from earlier representations of the Internet as a distinct (cyber) space. Interestingly, when considering the materiality of technical infrastructure of the Internet, the shift in attitudes to an intimate connection of the Internet with everyday life seems to have been based firstly on user practices, followed only later by significant changes in the technical configuration of the affordances available to users. Both Miller and Slater 2000 and Bakardjieva 2005, for example, present insights of Internet practices that appear both fascinatingly ingenious and intimately relevant to the everyday, offline, life of individuals and groups, while being based on very different technical materiality than the practices that researchers could observe only a few years later: in other words, users started appropriating and reconfiguring Internet affordances (and, while doing so, progressively assembling what in this dissertation is called lifeworld Internet) way before web applications explicitly designed to be appropriated and reconfigured became widely available and accessible to a general public. This is clearly visible by comparing, amongst others, the practices discussed in Miller and Slater 2000 and Bakardjieva 2005, on one hand, and in Miller 2011 and Miller and Sinanan 2014, on the other: only in recent ethnographic work pervasive SNSes, widespread multimedia realtime conversations and mobile phones as (sometimes primary) Internet devices are part of the normal technical infrastructure on which user practices rely; this transition happened during the timeframe of my fieldwork, as will be discussed in Chapters 5 and 6: the failure of the project analyzed in the former was also due to the assumption that users had already embraced Web 2.0 affordances, whereas throughout the interviews that inform the latter I observed how social network sites and Read/Write Internet practices had by then eventually become part of the daily lives of my research participants.

Whereas the actual practices and related affordances vary widely between contexts (my own fieldwork, for example, necessarily portrays traits of Internet use specific of young students in the privileged economic context—at least relative to global wealth distribution—of a western global city), and the relevance of specific technologies and narratives is debated in ethnographic studies of Internet users, an important common element of distinction between the beginning and the end of the 2000s decade is the availability of technical affordances designed to support (in various ways and to different extents) the practices of appropriation and reconfiguration that were already visible in earlier times. These affordances constitute what I grouped under the label of Read/Write Internet in Chapter 2, whereas the commonly known label for these is that of Web 2.0. This narrative was articulated within the hegemonic discourse through which 'the Web' has been experienced by users during most of the 2000 decade, and it has also informed the actual software engineering practices of successive versions of Web 2.0 sites and applications, as well as shaping mainstream media coverage: for these reasons, it is useful to start from the Web 2.0 discourse when tracing the historical trajectory of lifeworld and Read/Write Internet.

Both within academic and non-academic literature over the past decade, the exact traits and scope of Web 2.0 have been discussed in countless different ways, most of them informed by the broad definition proposed by O'Reilly (2005b) and his successive simplified versions (O'Reilly 2005a, 2006), while still providing a puzzling multitude of often irreconciliable focuses: the main point of agreement seems to be, indeed, that there is no substantial agreement (Floridi 2014, ch7). My aim here, therefore, is not to try to find a common ground between the innumerable existing interpretations nor to propose yet another one: rather, it is to highlight how the computational foundation of the core traits of O'Reilly's Web 2.0 manifesto relates to the analysis of computational agency at the core of my own research, as well as to the materiality of user experiences as described by research participants throughout my fieldwork.

In order to do so, it is useful to summarize here the key points of his attempt to shape an initial vision of what he then called Web 2.0; according to O'Reilly (2005b), the Web 2.0 is characterised by the following seven trends as they had been emerging in the early 2000s:

1. **The Web As Platform**: the web is now seen as infrastructure on which het-

erogeneous services are developed, rather than as a medium used mainly to publish websites and to transfer data, information and meaning.

2. **Harnessing Collective Intelligence**: by making it easier for users to contribute data and information, and by gathering and amassing data derived from user activity online, organizations can build vast pools of information, either for public use (e.g. Wikipedia) or to inform business decisions (e.g. advertising networks based on profiling of user interests).

3. **Data is the Next Intel Inside**: availability of ever–growing data sets (either proprietary or open) is seen as the informational layer upon which value–added web services can be built (e.g. geolocation services built on top of map baselayers).

4. **End of the Software Release Cycle**: thanks to the availability of software frameworks and processes to deploy updates to web apps in a quick and safe way ('rolling back' systems to a previous working state if an update introduces unexpected problems), organizations providing web services are able to constantly add new features, often multiple times per day, as opposed to the months–long release cycles distinctive of earlier software development practices. In practice, this is achieved through a combination of software engineering knowledge and computational management of software engineering practices, through 'meta–software' (software used to build, manage, test, deploy other software). An example is discussed later in this chapter when outlining computational management of software dependencies.

5. **Lightweight Programming Models**: by making available to developers programmatic interfaces to query and manipulate data (Application Programming Interfaces or APIs) rather than unstructured content on plain web pages, web apps seek to enable a simple interconnection of different data sources through minimal coding.

6. **Software Above the Level of a Single Device**: prompted by the growing

popularity of early Internet-connected devices predating smartphones and tablets, such as iPod music players and TiVo video players, this principle anticipates the demise of the historic centrality of desktop and laptop computers, giving way to a multitude of specialised and increasingly portable devices that use the web as common platform.

7. **Rich User Experiences**: in order to foster user engagement and provide 'frictionless' access to the ability to contribute content and information through web apps, an increasing attention to the design of user experiences is considered vital.

O'Reilly's business–oriented writing certainly introduces a significant bias in his analysis, as explicitly acknowledged by himself[5] when he stated that his aim was to establish Web 2.0 as a 'meme' that tried to summarise and promote key elements of the web that was evolving after the year 2000 dotcom bubble collapse. A comprehensive review of critiques to the problematic aspects of Web 2.0 as stemming from O'Reilly's seminal text has been discussed in Chapter 2; leaving aside for a moment O'Reilly's ultimate aim, however, a commonality between these seven traits is that they all imply focused (re)configurations of software code, whether to enable easier group collaboration, exchange of data or accessible user experiences. What is notable from a close analysis of O'Reilly's seminal text is the sparseness of focus on actual practices and genres, besides several mentions that serve as examples; his later attempt to simplify his own suggestion further highlights the computational foundation just discussed:

> Web 2.0 is the network as platform, spanning all connected devices; Web 2.0 applications are those that make the most of the intrinsic advantages of that platform. (O'Reilly 2005a)

The second part of O'Reilly's compact definition is then a substantive, albeit general, exemplification of what 'the network as a platform' means:

> [...] delivering software as a continually–updated service that gets better the more people use it, consuming and remixing data from multiple sources, including individual users, while providing their own data and services in a form that allows remixing by others, creating network effects through an

[5] As also argued in depth by Morozov (2013a).

> 'architecture of participation,' and going beyond the page metaphor of Web 1.0 to deliver rich user experiences. (O'Reilly 2005a)

Floridi (2014), citing this definition, consequently synthesizes it thus:

> So the Semantic Web is really the *participatory* Web, which today includes 'classics' such as YouTube, eBay, Facebook, and so forth. Just check the top twenty–five websites in Alexa, the web service that provides information about websites. (2014, ch7.5)

Actually reviewing the top twentyfive sites in Alexa[6], however, reveals a more nuanced picture: only ten of them are social networks or web services whose main participation element is configured around end users, whereas all the others are either ecommerce sites or search engines, whose participatory models are vastly different from that of Wikipedia, for example, as they involve computationally matching offers, purchases and logistics (in the case of ecommerce websites) or user search strings with relevance inferred computationally from continuous indexing and metadata harvesting of websites (in the case of search engines). Whereas the participatory trait has been widely accepted as characteristic of Web 2.0, this has commonly been interpreted as *participation between humans*, both in personal or public/political contexts. My argument, however, is that if we read O'Reilly's definition quoted above as a whole and together with the more articulated O'Reilly 2005b, and we consider that common implementations of Web 2.0 applications are software–based encodings of capacity for action that complement the largely hand–crafted adaptations of pre–Web 2.0 Internet affordances—as discussed at the beginning of this section—this participatory trait can more literally be read as *participation throughout a complex network of humans and machines*, based on the availability of code designed to support human practices.

Although this may seem semantic pedantry, I contend that it is in fact a useful interpretation as it allows to include in the analysis the centrality of computation as site of mediation between the social and the technical: when considering participation as an exclusively human concern (as exemplified by discussions focusing on user–generated content, collaborative practices and participatory politics), the dual role of machinic agency as enabler both of progressive practices and of large scale harvesting and processing of user data be-

[6] As of December 2015.

comes a distinct concern from the user practices per se, whereas including both human and non-human actants as part of a complex network of mutual interdependence allows to highlight:

- the role of technical affordances in enabling, defining and limiting possible spaces for user agency
- the role of humans and organizations providing software services, whose agendas are often in conflict with those of their services' users
- the role of other organizations (such as governments and providers of Internet advertising services) as further parties to Read/Write Internet user practices
- consequently, the possibility to interpret technical affordances that users encounter in everyday life as configured according to specific rationalities, among a range of possible alternative ones, each inscribing in technical forms different spaces for user agency

When analytically considering computation as a site of mediation, the interdependence of distinct models of user agency and information politics becomes integral part of a more complex understanding of Read/Write Internet (Jordan 2015), bringing the practices that violate user freedoms, privacy and self–determination to the same level of analysis as the progressive practices that a Read/Write use of the Internet could enable: in other words, allowing to consider that each writing practice (such as, within the domain of lifeworld Internet, exchanging messages and photos with a relative, for example) can be countered by reading practices that may be hidden from end users because of the computational mediation that sustains them (for example, the capture of conversation metadata or data by the service provider for ad targeting purposes or by a government agency for 'intelligence' purposes), while effectively framing the users' computational agency (for example, by causing 'chilling effects' whenever users may doubt whether their private conversations are being intercepted by third parties without consent or without full and clear disclosure).

When applied to the analysis of my first stage of fieldwork, this framing of Read/Write Internet will be used (in Chapter 5) to examine how the project failed also because of a missed articulation of its computational complexity: actants that saw themselves as users (managers, departmental coordinators) attempted to negotiate a primary role alongside that of the core audience of the project (the university's students); knowledge formalizations inscribed in the software libraries and informing the best practices of the time conditioned the possible configuration of the code being developed internally; the lack of substantial use of external Web 2.0 applications among students at that stage negated the usefulness of an approach that attempted to let students integrate external data in personal 'mashups'. Similarly, most of the user practices emerging from the accounts of my fieldwork participants (Chapter 6) highlighted the central role of the 'walled gardens' of major SNSes as sites containing and informing everyday computational agency of these students, and the struggles of the few students involved in active reconfigurations of software code behind the affordances they used in everyday life revealed the implicit limits to the advanced traits of computational agency—the capacity to reconfigure an user's computational environment itself beyond the options for configuration and appropriation available through the software's design.

4.3 Deconstructing the *telos* of progress

The analytical stance introduced in the previous section highlighted that, historically, technical form ('Web 2.0') has followed user practices of appropriation and reconfiguration, until it became the normal computational context within which user agency was articulated, and that, when considering the letter of O'Reilly's proposal of Web 2.0 as 'the network as a platform', end users' computational agency can be integrated in the complex network of interrelationships between human and machinic actants and of the agendas negotiated by the parties involved, through the layer of computation.

In this section I deconstruct the telos of progress inscribed in mainstream discourses related to the Internet, with the aim of highlighting its disconnect from the technical materiality of Read/Write Internet, as framed through the two traits just articulated: my argument is that the hegemonic discourse surrounding lifeworld Internet (and Internet in general, although this is beyond the focus of my analysis), by employing a telos of progress (and by encoding it in an intuitive narrative of 'versions', as discussed in the second part of this section) has successfully established in mainstream discourse the acceptance of technical form as *leading* improvements of practices rather than *following* them and the human needs that are connected to these. This epistemic reversal operated through narratives of technology development is particularly visible—again, in a field of research outside of the focus of my dissertation—in the ICT4D (ICT for development) discourses: see Slater 2013 for a comprehensive critical overview. Although the consequences of such reversal may be less immediately visible—and and generally less immediately devastating—within the domain of lifeworld Internet on which my dissertation is focused, this reversal contributes, in this context, to the mainstream acceptance of a largely one–dimensional narrative of 'progress', as criticized by Berry (2014) and Morozov (2013b): alternative rationalizations (such as those analyzed in Chapter 8) hence become reactions to and contestations of an established hegemonic narrative and technical form rather than being part of a diverse foundation for computational agency that includes multiple, alternative architectures responding to different needs in different contexts. An example of convergence towards an hegemonic architecture is discussed in the third section of this chapter, where the historical transition towards a client/server (rather than peer–to–peer) architecture is analyzed, alongside its implications for users' computational agency; hacker responses to this closure of space for alternatives are then discussed in chapters 7 and 8.

Whereas the '2.0' version number visibly promoted through the Web 2.0 label successfully synthesized in an immediately understandable way a narrative of a new stage of development—a *reset* or new start of the web, based on a computational platform supporting human practices—a closer look at the materiality of technical change of the Read/Write Internet discloses practices that contradict the mainstream rhetoric of a constant evolutive progress towards better, faster, more personalized user experiences enabled by web and mobile apps. Contradicting the very idea of a new phase of Web infrastructure based on the ability to easily perform two–way interactions, trying to precisely locate in time the inception of Read/Write Internet may appear to be a frivolous and ultimately pointless exercise: on one hand, the Internet has been since its early times a fully Read/Write medium, at least if considering the ability of each node to be by design both a server and a client at the same time, and therefore to potentially consume information from other nodes and to send information to other nodes, thanks to the packet switching technology on which the lowest layer of Internet infrastructure rely; moreover, O'Reilly's 'meme making' text (O'Reilly 2005b), in order to discuss the innovative properties that had allowed Web 2.0 to become a clearly identifiable stage of the evolution of the Internet, builds on a multiplicity of traits, each in turn implemented in practice through countless alternative technical implementations, often being created at the same time by mutually unknown developers: the Read/Write Internet, like many phenomena involving large populations, has effectively come to be through an increasing diffusion of practices and technological innovations which, over a few years, became established and started being relevant to larger groups and having a tangible impact on people's lives, on business processes and on society at large.

Likewise, phases of development and even widely accepted milestones of Read/Write Internet (e.g. the introduction of entirely new types and formats of devices, such as smartphones and tablets) can rarely be clearly delimited, as they often involve multiple layers of tangible and intangible artifacts (software, hardware, infrastructure) as well as multiple layers of agency. To illustrate this with an example: designers and developers at a large company may have been working for months on a new product (such as the first iPhone) by the time the company's marketing apparatus starts briefing journalists and building public awareness through advertising; app developers may get access to documentation allowing them to build software for the new product at a yet different stage; and once the device is finally available for purchase (a phase which often starts at widely varying moments in different countries), its uptake may take months to build a sizable user-base and to effectively have an impact on users' lives and on social processes; acknowledging and properly analysing the characters and nature of these effects may, in turn, take even longer as changes accumulate and emerge over time, and may for a while be enacted simply as variations of pre-existing practices (Bolter and Grusin 1999) before fully taking shape and being acknowledged.

Even though the marketing rhetoric of major media and technology enterprises routinely celebrates 'innovations' and 'breakthroughs' at launch events[7] and press briefings, the process leading from ideation to uptake of most Internet technologies is rarely linear and it often involves reconfigurations of designs, public perception and expectations at various stages; moreover, competing products may influence each other's development in different ways, and even the introduction of substantial innovations is often staggered on purpose in order to enable planned obsolescence (Fuchs 2014a; Maxwell and Miller 2012). This is particularly relevant for hardware products, but software development may as well be kept under a controlled schedule in order to release only later on features that make devices obsolete when they don't support the new software features being introduced.

Moreover, and perhaps more importantly, both innovative and routine changes to Internet affordances can be introduced in stages by design: web apps and social network sites often introduce new features through 'A/B testing', whereby different users see distinct variants of the app's user interface, while data is gathered on metrics such as conversion to sales, ability to reach user interface goals, etc. associated with each interface variant; this data then influences how features are further developed and introduced as stable parts of the user experience. Changes to Internet affordances also reach different groups at different stages according to material factors such as location (staggered introduction across different international markets), income (allowing or delaying the possibility to purchase latest and more expensive devices), different patterns of discovery (through peers, via marketing efforts, from mainstream press, etc.), relevance and cultural norms (a social network feature may quickly become widely used by users in certain age groups whilst being seen with suspicion, for example, by younger or older users).

Having discussed how changes happen mostly gradually, are often composed of a multiplicity of interconnected factors and are easier to identify once they have been informing user practices for some time and at scale, a further necessary observation is that even in an historical phase characterised by the 'end of the software release cycle'[8] and by continuous feedback loops between usage and refinement of software, an analytical distinction is always present between design and development of technologies, on one side, and user prac-

[7]Besides rarely marking a true pivotal moment in itself, launch events and major announcements are also subject to external scheduling pressures: for example, global recurring technology events such as the yearly *Mobile World Congress* and *Consumer Electronics Show (CES)* trade shows, the *Apple Worldwide Developers Conference (WWDC)* and *Google I/O* developer conferences, as well as sales patterns close to December holidays often predictably dictate when new consumer Internet devices are going to be presented for the first time to the public.

tices, on the other. Even when user actions are monitored, tracked, analysed and assessed in order to inform data–driven design choices for further iterations of a web application or a device, user agency is ultimately filtered through the knowledge models of the designers/developers and the business objectives and strategies of the companies behind the technologies being developed and the infrastructure across which data and information is transmitted. This is a notable example of jarring disconnect between a purported increase of user participation enabled throughout the Read/Write Internet and actual agential outcomes.

4.3.1 Narratives of versions

As discussed in the previous section, the historical development of Read/Write Internet is characterised by a multiplicity of factors operating according to different logics, in different directions and at different speeds, only rhetorically subsumed within a unifying narrative of progress aimed at promoting the consumption–oriented idea that successive versions of apps and devices are 'better' than the previous ones and therefore desirable. An analysis of mainstream discourses surrounding the Read/Write Internet highlights how this telos of progress is intimately encoded through narratives of versions.

The most widely used label for Read/Write Internet technologies—Web 2.0— embeds prominently in the name itself a rhetoric of progress, evolution and distinction: the circulation of such label introduced in public discourse the idea of software–like 'versioning' (e.g. 2.0); whereas an integer number–based versioning scheme is commonplace in lay contexts (for example, as a way of distinguishing successive iterations of a document by embedding a progressive integer number in its filename), the concept of version numbers composed of major, minor and patch numbers separated by dots (e.g. 2.0.0, see Preston-Werner 2013) is pervasively used by developers as a fundamental component of modern software engineering practices. The '2.0' part of the Web 2.0 label is therefore immediately recognizable as a version number specific to software artifacts, and it is aptly chosen to mark a sharp distinction from any lower (one–dot–something) version numbers: already in its first occurrence (DiNucci 1999) it is used to denote traits and logics of a 'new

[8] See O'Reilly 2005b, trait n.4.

phase' of the Web, clearly distinct from what readers of DiNucci's articles may have been experiencing at the time.

Tim O'Reilly's rhetorical influence and business–oriented aims cemented the version–based label as a signifier of a set of technologies and approaches that in his vision were starting to reshape the way the Internet was being weaved into users' daily lives. Allen (2013) articulates this 'discourse of versions' by analysing how the introduction of version numbers

> [...] created a 'history' of the internet, constructing what it claimed to describe, and influencing our collective, public understanding of the internet through this historicization as much as in any other way. (Allen 2013, p261)

Through his analysis, Allen traces how the introduction of a '2.0' version number led to the creation of the retronym 'Web 1.0' (already used in DiNucci 1999 and O'Reilly 2005b) to refer to the traits of 'pre–2.0 web'; Allen also reviews the opinions (such as that of Berners-Lee 1998) according to which Web 2.0 is what the Web was supposed to be since the beginning, introducing the retronym 'Web 0.0' to refer to Berners–Lee's original design, and arguing that the choice of 2.0 was meant to clearly denote a 'reset', as well as a phase of wider understanding of the web as an articulated artifact in close relation with contemporary cultural life:

> The particular nature of the wrong direction of the web, which some use to sustain the idea that Web 2.0 was a 'reset', is best understood by analyzing the problem of 'design' and online technologies. Prior to the web, the internet was constructed according to the aesthetic and functional conventions of computing, known only to a few. Part of the attractiveness of the web, by contrast, was that it relocated online activity into the normative, but well–understood, space of traditional media design, thus making it technically more accessible and culturally more legible. (Allen 2013, p267)

Through the mainstream diffusion of Web 2.0 discourse, version numbers have since been used widely to refer to purported new stages of evolution in other fields of human activity (education, healthcare, urbanism and so on), with a 2.0 version marking 'resets' from previous customs, and with higher version numbers (3.0 and higher) typically used to denote future stages of evolution (for example in Swan 2015, discussing potential disruptive future uses of blockchain technology beyond bitcoin monetary exchange).

Although it may be considered a seemingly incidental rhetorical artifice, the adoption of version numbers from software engineering can therefore be read as an integral part of the materiality of Read/Write Internet. On one hand, it helped to promote the idea of a new phase, after the disappointment of the first dotcom collapse, of development of technologies qualitatively better positioned to match user expectations (and therefore to generate value), according to a deterministic telos of constant progress, reflecting in increased opportunities for the relevant stakeholders. On the other hand, it implicitly diminished the value of user practices predating the '2.0' pivotal moment: whereas ethnographic research has unearthed the intricate and very personal processes through which individuals make sense of technology within their lifeworld horizon, making use of whatever technology is at hand, the focus of mainstream narratives on constant improvement and progress enabled by computation has relegated these human practices—that computation is supposed to sustain—to the role of accidental instances. A powerful, computation–enabled rationalization subsumes individual differences, closing the spaces available for computational agency through alternative rationalizations, and encoding in mainstream Internet technology what Berry, referencing Adorno, describes as the dangers of 'identity thinking', when suggesting the urgency of 'a project exploring in what sense critique and critical thought can address the computational' (Berry 2014, pp12–13):

> Here identity thinking is understood as a *style of thought* that aims at the subsumption of all particular objects under general concepts, and as a result the particular is dissolved into the universal. The distance between computational knowledge and reality is entirely closed when we think we have succeeded in framing reality within these computational categories and by means of computational methods. This is a dangerous assumption, as it is a short step towards new forms of control, myth and limited forms of computational rationality. (ibid., pp12–13)

4.4 Topology and agency: client/server over peer–to–peer

Read/Write as a *modus operandi* of a communications medium entails both practices (two–way communications and information exchange) and specific configurations of the technical infrastructure that supports these practices.

Although the modern Read/Write Internet started taking shape in the early 2000s thanks to innovations that supported the use of the Internet as a two–way medium, these innovations were not only responding to existing practices of appropriation and reconfiguration of pre–Web 2.0 technical form—as discussed in the previous sections—but were also building upon lower layers of infrastructure that supported Read/Write communication since the origins of the Internet in the 1960s. Analytically, it is useful to discuss two partially overlapping but distinct layers: on one hand, a client/server versus a peer–to–peer (p2p) mode of information exchange; on the other, Read/Write versus (mostly) Read–only practices. These two layers operate at different levels: a client/server infrastructure, while potentially supporting both Read/Write and Read–only practices, is based on a 'star topology' whereby a server[9] acts as a central coordination point between a multitude of clients. Its function can range from that of a neutral *gateway*, facilitating interactions between the client and itself and between clients, to that of a *gatekeeper* that actively orchestrates which kinds of exchanges and interactions are permissible between itself and clients, and amongst clients. A p2p infrastructure, on the contrary, is usually based on a 'mesh topology' that puts each node in contact with several other nodes, each of which is in turn in contact with several other nodes and so on, making it possible to route interactions between any two nodes through the mesh, without relying on central coordination operated by special nodes (the 'servers' in a client/server architecture).

It is important to note, however, that the actual material configuration of any infrastructure or portion thereof is almost never as unequivocally configured as per the abstract definitions: complexities introduced by different types of considerations determine the actual shape of infrastructures. Famously, for example, the Skype communications system relies on a client/server model to manage security–sensitive functions such as user authentication, accounting of calls and charges, payments and other monetary transactions, while orchestrating a p2p infrastructure between user nodes to route messages and audio/video streams; furthermore, some of these nodes operate as 'supernodes', coordinating subsets of neighbouring peers[10].

When considering configurations of practices, both infrastructure topologies (client/server

[9]In practice, most of the times such server is actually a set of servers, each carrying out different functions, although from the end user's point of view this is often an irrelevant implementation detail: a request for some kind of information is sent to a 'server' and a response is sent back to the client.

[10]A detailed and up–to–date technical explanation of this hybrid architecture, highlighting the roles of different types of nodes and motivations for this choice of architecture, is published by Skype itself: see Skype 2015.

and p2p) can support both Read/Write and Read–only modes of communication: it is mainly the topology that differs, but the intended outcome is to exchange information between endpoints, whether bi–directionally or mainly as a one–way flow[11]. A client/server topology, however, assigns to parts of the infrastructure a gateway role, introducing the possibility of exploiting this technical asymmetry by turning it into a power asymmetry (be that to filter, monitor, record communication, to impose fees for access to specific content, to give higher or lower priority to traffic to or from specific sources or of a specific kind—such as limiting bandwidth for bittorrent streams—or to enforce specific communication modes such as Read–only); a peer–to–peer topology, on the other hand, may include malignant peers who try to enact similar control on the content flowing through them, but a sound design will allow other peers to 'route around' problematic nodes: see Gilmore, in Elmer-DeWitt and Jackson 1993, although recent developments (e.g. blockchain technologies such as those used for the Bitcoin currency) are questioning this 'pre–computational' faith in the eventual consistency of large networks and are highlighting the complexity of maintaining integrity of consensus even in very large mesh networks when groups of users control large pools of resources close to 51% of the network extent (Bradbury 2013; Perry 2012a).

Historically, peer–to–peer was the original *modus operandi* of the Internet, by design: a handful of research and military computer systems were permanently connected to early Internet links, always on and able to send and receive information as required by their users (Abbate 1999, ch.6). One of the first practical uses of the early Internet—email exchange—relied on this peer–to–peer infrastructure to enable two–way information flows between senders and recipients. The basic principles of email exchange between end users are essentially the same nowadays as they were in 1982 (Klensin 2001, 2008; Postel 1982), but a variety of factors[12] have reshaped profoundly the ways in which email infrastructures are currently deployed, although the intended outcome is still often simply the enabling of asynchronous two–way exchanges between senders and recipients.

[11] An accurate description would need to always include *some form* of bi–directional exchange, as clients obviously need to contact servers in the first place to request the start of a one–way broadcast, but the point here is to focus on the substance of information flow rather than on the technical details involved.

[12] E.g. the need to control huge volumes of spam and malware–carrying email messages, value added services built on top of basic email infrastructure, state censorship, etc.

Minar and Hedlund (2001, pp9–15) outline how the client/server topology became dominant in the 1990s, through a combination of commercial considerations, security reasons, design limits and physical constraints. The surge of spam messages since the mid–1990s, disrupting the (until then) fundamentally open infrastructure of Internet email[13], is presented as a driving force for the switch to a centralised email infrastructure, whereby specialised email servers accept, exchange and store messages on behalf of end users, applying a range of machine learning techniques to identify and filter spam messages (and, more recently, malware and scam/phishing emails). Besides the specific case of uncooperative actors disrupting through unsolicited email the peer to peer origins of email protocols, Minar and Hedlund highlight how it is generally the lack of accountability inherent in the original Internet infrastructure to allow for a wide range of behaviours to be enacted over the network:

> The lesson for peer–to–peer designers is that without accountability in a network, it is difficult to enforce rules of social responsibility. [...] Technologies for accountability, such as cryptographic identification or reputation systems, can be valuable tools to help manage a peer–to–peer network. There have been proposals to retrofit these capabilities into Usenet and email, but none today are widespread [...]. (Minar and Hedlund 2001, p11)

Minar and Hedlund's main point is that the Internet infrastructure was designed with a specific, relatively small scale use case in mind (information exchange between cooperating research organizations), and that even though it has undergone a vast evolution over the past decades—both in terms of protocols and capacity—it is still fundamentally limited by the original assumption that no uncooperative peers would be part of the network; or, more appropriately, that lack of cooperation from a technical point of view (e.g. in the case of failure or erratic behaviour of a node due to software errors) could be 'routed around' until fixed, whereas lack of cooperation by way of purposefully malignant human behaviour (such as the case mentioned by Minar and Hedlund of the first recorded commercial mass unsolicited email posting — Canter and Siegel's 'Green Card' usenet posting: ibid., pp10–11, and—as as a contemporary concern—the trust and security issues related to blockchain technologies, as mentioned earlier in this section) could be addressed through

[13] In 2014, spam was estimated to account for 66.76% of worldwide email traffic, according to Vergelis, Shcherbakova, et al. 2015.

a combination of moral arguments (condemnation of the uncooperating behaviour to discourage its reoccurrence through peer pressure) and technical measures (such as the termination of an offending user's account on the abused system).

As the Internet quickly evolved towards a commercial network potentially open to any person and organization, these original assumptions, inscribed in the fundamental design of the Internet's lowest layer of packet–switching networks, were not tenable anymore; their consequences, however, deeply affected the development of Read/Write Internet over the following years: oversight of the Internet at a very large scale could be ensured only through highly centralized control of key information flows, in order to limit the potential for abuse of the peer–to–peer infrastructure: as this wasn't designed to accommodate security and trust at its lowest layers, these layers needed to be protected from direct manipulation by arbitrary users, leaving only higher–lever abstractions available to untrusted parties.

The historic switch to a client/server topology during the 1990s and the later diffusion of peer–to–peer architectures can be read as successive attempts to realign the original (peer–to–peer) Internet architecture to the transition towards an increasingly Read/Write Internet. Ironically, the infamous quote by US Senator Ted Stevens 'The Internet [...] [i]s a series of tubes' (Stevens 2006), although part of a weakly argumented (Felten 2006) speech about net neutrality, captures the core tenets of the early Internet, focused on information flow with the basic aim of delivering information between end nodes. As Read/Write practices became widespread, however, interactions between users became more sophisticated, requiring more than plain transmission of discrete units of content (such as web pages or video files) between endpoints: instant messages, content shared selectively with friends or relatives, data and metadata captured by sensors or sent to actuators (such as IoT devices operating in the home environment) all require distinct types of data exchanges (e.g. real–time, two–way exchanges—or many–to–many for group chat—for instant messaging applications; secure, resilient and low-latency transmission for IoT data—for example, to ensure that a fire alarm is promptly propagated as soon as smoke is detected in a home, and that nobody could interfere with data exchanges, for example by listening in to a baby monitor via unencrypted WiFi connections). Whereas most of these exchanges involve direct communication between individuals, often from their homes, the layer of control developed to protect users and service providers alike from malignant actors when the Internet became open to arbitrary users forces most of these exchanges to happen through client/server topologies.

Additionally, a very banal but long–standing fundamental technical limitation of the core protocol of the Internet's infrastructure (IP protocol version 4: Postel 1981b) further determined the historical need for gateway servers: the number of devices directly connected to the Internet with an unique network address ('IP address') is limited by design to less than four billion[14]: whereas this may have legitimately seemed a sensible limit when the IP protocol was designed, the rate of growth of the commercial Internet throughout the 1990s highlighted that the available address space would have become unsuitable in the longer term; although a new version of the IP protocol (IPv6: Deering and Hinden 1998) was created to overcome this (and other) issues, its adoption has been hindered by the vast legacy of connected devices unable to use the newer protocol. In order to avoid exhausting the available address space, several techniques were developed; the most common (Network Address Translation: NAT) allows to use a single public IP address to provide Internet connectivity to practically unlimited individual devices, and this technique is in fact widely used for both domestic and business networks connected to the Internet; a major limitation imposed by this configuration, however, is that direct connections between arbitrary devices behind different NAT private networks require some form of external coordination (such as coordination servers, or 'supernodes' as often used in p2p networks). Many of the projects discussed in Chapter 8 explicitly deal with this single major issue, which—although an historical accident—has deeply conditioned much of mainstream Internet architecture and related practices since the early years of public Internet.

[14]The actual theoretical limit, 2^{32}, is slightly larger than four billion, although implementation details (reserved address ranges, etc.) effectively make the number of IP addresses usable by endpoints lower than that.

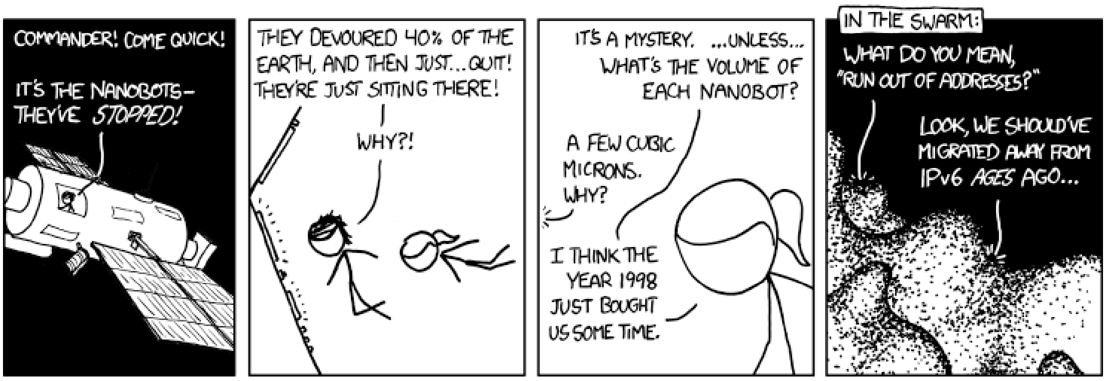


Figure 4.1: 'Nanobots' (from xkcd comics)
`http://xkcd.com/865/` — CC BY-SA 2.5. The number of IP addresses that can be used throughout the public Internet using the IPv4 protocol (less than 2^{32}, or about 4 billion—and, most importantly, less than one per human being) constitutes a major limitation to the design of core Internet infrastructure, which has effectively contributed to the dominance of client/server configurations and computational intermediaries to establish reliable links between devices behind routers using 'network address translation' (NAT). The IPv6 protocol, formalized in 1998 (Deering and Hinden 1998), among other network-level improvements could allow to attribute to each personal or IoT device its own IPv6 address potentially reachable without the need of computational intermediaries, although in practice only experimental projects rely on such capabilities to implement unmediated peer–to–peer configurations, due the limited adoption of the IPv6 protocol so far.

Moreover, commercial assumptions about content consumption patterns have also shaped another important trait of the topology of mainstream Read/Write Internet: namely, that content would need to flow from specialized providers (e.g. video hosting services) to end users much more than between users, as discussed by Minar and Hedlund:

> ADSL and cable modems assume asymmetric bandwidth for an individual user. This assumption takes hold even more strongly inside ISP networks, which are engineered for bits to flow to the users, not from them. The end result is a network infrastructure that is optimized for computers that are only clients, not servers. (Minar and Hedlund 2001, p15)

As for the limitations of the IPv4 protocol, this design decision could have made sense when processing and hosting multimedia content was very resource–intensive and expensive; in recent years, however, decreasing costs of computer storage made self–hosting personal media potentially feasible on cheap storage devices connected to home networks, although sharing (for example, with family) such media items is often unpractical due to the limits imposed by the asymmetry of transfer speeds described above, which only very recently are starting to be overcome in some urban contexts where bidirectional high–

speed connections for domestic users have been made possible by new fibre optic technology. For over twenty years, however, assumptions about highly asymmetric upload and download speeds have conditioned the physical architecture of mainstream Internet, making centralized media hosting services such as Flickr and YouTube extremely popular and limiting the options available for easy peer–to–peer sharing of personal content (as opposed to general–purpose p2p file sharing networks, whose architecture can be simpler as it often does not need to take into account security and privacy issues that users may consider essential when dealing with personal content). The long–lasting implications of this configuration of topology on end users' computational agency are addressed, as well, by some of the alternative rationalizations examined in Chapter 8.

Reviewing this analysis of competing architectures of mainstream Read/Write Internet, it is useful to note that the mainstream client/server configuration is, technically, largely implemented on top of an underlying peer–to–peer infrastructure, some traits of which are reconfigured to present a convenient and familiar interface for common tasks, requiring little or no technical knowledge from end users. Although some of the current alternative configurations provide viable substitutes to centralized services, they often require specialized appliances (for example, micro–servers to be connected to a home network such as those needed for the FreedomBox project[15]) and a degree of technical knowledge simply not available to most common users, limiting the potential public of configurations that try to reproduce the peer–to–peer infrastructure of earlier internet on top of the limitations imposed by the mainstream client/server topology: as this in turn is a reconfiguration of the underlying peer–to–peer infrastructure, the computational complexity involved in the coordination of this double layer of translation of topology (from peer–to–peer to client/server back to peer–to–peer) requires technical, epistemic and policy adaptations that have not been tested by independent groups of developers at a large scale so far.

Finally, from this discussion of a dichotomy of competing topologies one should not rush to conclude that a pure peer–to–peer topology constitutes a sort of pure 'ur–configuration' of the Internet, in a similar way to Turkle's recent work about authenticity of interpersonal

[15] `https://freedomboxfoundation.org/`

communications (Turkle 2011), critiqued by Horst and Miller (2012a) who highlight that in–person communication in any culture always involves a range of acts of mediation, even in the absence of digital technologies. Similarly, the physical topology of infrastructure is only one of the many layers involved in the configuration of computational agency over the Internet, and it is essential to remember that the early Internet served vastly different scales and purposes than today's commercial Internet, making a direct comparison of topologies and power configurations between then and recent years less useful than an analysis of the implications of the materiality of current Internet on current models of computational agency—a task that I undertake throughout this dissertation. Moreover, client and servers have always been part of Internet architecture even if in much different shapes: striving for an a priori all–or–nothing redecentralization can be as problematic as decontextualized 'information wants to be free' claims (as articulated in Jordan and Taylor 2004, pp168–169), and may—conversely—confound the strengths of redecentralization efforts. On the other hand, a critical look at campaigns such as the one run by Google in recent years based on the slogan 'The internet is what you make of it', and current proposals by Facebook to provide 'free' Internet connections in developing countries whilst effectively retaining a tight central control over which content can be accessed (Custer 2015; Murthy 2015), highlights the complexities that can be easily hidden within the technical materiality of Internet infrastructure: articulating this complexity and questioning the ability for configurations that retain their centralized topology as a legacy of earlier technical, commercial and policy issues to sustain a meaningful computational agency of end users in lifeworld Internet has been my aim through the present section.

4.5 The *computational turn* of the Internet

Suggesting that there has been, in the recent history of the Internet, a 'computational turn'—a transition to a much higher reliance of the operativity of the Internet on computation than at any other previous stage—assimilable to those of fields or disciplines traditionally not relying on computation (e.g. as in the development of *digital humanities*: Berry 2011) may sound disingenuous, as computation and computational devices constitute the essential material fabric of the Internet, rather than being one of several possible modes of existence.

However, as will be discussed in this section, around the years 2010–2011 a combination of conditions developed over the previous decade triggered a noticeable shift in the dominant shape of the Internet, leading to the establishment of configurations that rely—proportionally—much more on computational power at all levels (infrastructure, server–side and client–side) than at any time in the past history of the Internet. Although the core pivotal elements of this computational turn are technical, both the conditions under which they took shape and precipitated, and the effects they had on the user and developer experience of Read/Write Internet and on computational agency are mainly social and political, and therefore of central relevance to the analysis of computational agency which is at the core of my dissertation.

The point here is not to identify a precise juncture, whose extent and velocity could be the subject of future research: it is, instead, to deconstruct a shift that has visibly occurred across the main timespan on which the empirical materials used throughout the dissertation are focused (2005–2015): the factors leading to the computational turn of the Internet operated across this timespan, making the Internet in 2005 and the practices that it sustained at that time markedly different from those of the Internet in 2015.

As outlined in Chapter 2 and discussed earlier in this chapter, early narratives of the Internet privileged the role of transmission of information and data, in turn sustaining the rhetoric of new possibilities to freely exchange information and to directly reach people and resources that would have been harder to reach without the Internet, as well as underpinning the Web 2.0 promise of enabling individual users to publish personal content with little or no need to know how to use complex tools. Whereas (digital) *transmission* of information was (and to a certain extent still is) the leading narrative of Internet form, it is almost a platitude to remark that transmission itself, as a digital exchange, relies intimately on *computation*: from the code that routes packets across the network between the endpoints (computers) of any low–level data exchange, to the software needed to prepare data for transmission on a server and to display it on the client, the material Internet can only exist through the execution of code.

Yet, the computational turn of the Internet marks a profound change in the ways in which computation itself is enacted. On one hand this entails the introduction of entirely new computational artifacts and actants ('compactants': Berry 2014): the ones more centrally relevant to the reconfiguration of spaces for computational agency within the domain of lifeworld Internet are discussed in detail below. On the other hand, this change also includes a marked shift towards machine–to–machine exchanges, where no human actors are ever involved, and only data and requests to process data are sent, with more data returned or generated and stored as result of the processing requested by the other machine(s): while often not visible to users in their everyday practices, these background exchanges between machines affect the shape and space for agency of the affordances available to users. Moreover, although the recent increased visibility of new classes of non–human actors can be attributed in part to the 'Internet of Things' rhetoric promoted by vendors of IoT devices, software and services, the diffusion of Internet-connected sensors, actuators, appliances and robots in the public sphere and in the domestic environment is rapidly changing mainstream understanding not only of what kind of information is exchanged across the Internet, but also by which classes of actors. These reconfigurations of computation and of its understanding started to become visible in the second phase of my fieldwork (Chapter 6) and constitute the terrain of struggle of the hacker interventions examined in Chapters 7 and 8: both users and developers, to various extents, were intent in exploring the question of which models of computational agency are possible within the changing context of the computational turn of the Internet, and in turn, which agendas these models of computational agency can sustain, whether in the case of individual users, of classes of users in specific contexts, of service providers, media industries, advertising industries, etc.

Interestingly, whereas the computational aspect of information exchange became increasingly relevant through the computational turn also thanks to the expansion of Internet–connected actors to the non–human (and hence essentially computational) ones just discussed, from the point of view of end users the narratives analyzed in Chapter 6 highlight a parallel shift in the opposite direction: the sense of digital and computational mediation still dominant in user accounts of earlier studies, even successive to the 'ethnographic turn' of Internet studies discussed in Chapter 2 (Bakardjieva 2005, Miller and Slater 2000), is replaced by an increased backgrounding of the digital affordances themselves, as devices become less intrusive (both physically—as sizes increasingly shrink down to ultraportable computers and to smartphone and tablet formats—and in terms of the care they need to operate and to connect to the Internet), whereas the (mostly entirely non–digital) everyday life concerns of users increasingly become the focus of their (digital) practices sustained by software and devices quietly and often invisibly gathering, processing, transmitting and presenting information.

4.5.1 Key traits

Given the centrality of the analytical framework of computational turn of the Internet to my research, in this final part of the chapter I outline its key traits, discussing their role in the reconfiguration of agency within the domain of lifeworld Internet.

Given that the computational turn of the Internet is a very recent phenomenon, not all its aspects have been analyzed in depth in scholarly literature. Some traits and their social and political consequences, such as in the case of cloud computing, are sufficiently well understood thanks to recent research (Berry 2014; Jordan 2015; Mosco 2014). However, I have relied more decisively on developer discourses in order to establish the relevance of the traits discussed here: a close review of 'engineering blogs' can provide useful information of the choices of technologies and related motivations happening within large corporations and small startups alike; occasional contributions to computer science publications by engineers responsible for infrastructure projects discuss in more formal ways findings from projects aimed at optimizing internal processes; I have also reviewed public discussions (e.g. on the Hacker News website and through the *Zeitgeist* of themes covered at major tech conferences) about these disclosures of technical choices in order to gain an understanding of the assessment of relevance across a broad spectrum of developers: the traits discussed here as central to the computational turn of lifeworld Internet, whereas not the only ones that contribute to the distinction of post–2010 Internet from earlier configurations (a brief overview of some traits that were excluded and reasons for that is presented below), are those which through the sources outlined above were portrayed as closely relevant to the experiences of my research participants and to the hacker practices analyzed in Chapters 7 and 8: for each trait, I present below a brief overview and a discussion of their implications on computational agency within the focus of my dissertation. One common character, specifically, is that all the four core traits are shown to be enabling, on one hand, an increasingly efficient centralization of lifeworld Internet within the control of a small group of corporations, thanks to automation and efficient use of resources, therefore expanding computational agency of private actors at the expense of end users (although the assessment of this is complicated by the fact that—as discussed through the analysis of user accounts in Chapter 6—users may *perceive* an actual expansion of the variety and quality of affordances at their disposal, whereas the implications of the centralization of computational capacity on information politics is often opaque to end users themselves). On the other hand, all the traits analyzed are also shown to be enabling more efficient development of alternative rationalizations, whose details are analyzed in Chapters 7 and 8.

Three major emerging trends are not included in my discussion, for different reasons. *Commoditization of server farm technologies* is an established practice, as major Internet/advertising companies have been shifting towards producing and assembling their own hardware, optimized for specific computational tasks, as discussed by Jordan (2015) for the case of Google, and as manifest through Facebook's own Open Compute project[16], through which specifications, designs and documentation of Facebook's custom–built server farm hardware is made available for reuse. Although increasingly relevant also for smaller organizations (Sverdlik 2014), and adopted to optimize the efficiency of computation at large scale, this practice is not (yet) as pervasive as the key traits discussed below. Direct control over hardware is, however, an increasingly relevant issue for hackers developing alternative rationalizations: a discussion of the implications of current trends on computational agency is developed in Chapter 8. *Big data* and *machine learning*, likewise, are an increasingly common focus, either as primary aim or as supporting technologies, of startups and are practices that rely vitally on efficient, large scale computation and data storage (Kitchin 2014, ch.5); their role, or possible role, within alternative rationalizations seems still unclear, however, and consequently while certainly affecting user practices (for example through data-driven optimization of the user experience of major social network sites), they fall outside of the focus of my research, which aims at identifying computational strategies that not only support the infrastructure of mainstream lifeworld Internet but can also be meaningfully appropriated by hackers involved in alternative rationalizations. Finally, the *Internet of Things* is a field on which startups and experts' interest are increasingly focusing, but its (relative) relevance in lifeworld Internet contexts is very recent (for example, through the introduction of Internet–connected smart lighting systems and home heating control systems) and as such it was not part of the daily experiences of my fieldwork participants. Nevertheless, challenges related to control over the hardware of Internet–connected 'things' and to ownership of data produced by them will likely be interesting research topics for scholars who wish to explore developments of computational agency within the personal sphere in the coming years.

The key traits highlighted by my review of seminal scholar literature and developer discourses are discussed in the next four sections.

4.5.1.1 Computation as a service: who owns computational capacity?

In broad terms, cloud computing can be conceptualized as *computation as a service*. In the early years of the Internet, companies, organizations and individuals wanting to run Internet–accessible software such as web apps and their backend infrastructure (databases, monitoring services, authentication services, etc.) would often have to directly procure and manage (physical, or 'bare-metal') servers, investing in the related skills needed to operate them. By making computation capacity available as a service, instead, cloud computing providers allow organizations to only pay for the capacity required at any given time, typically with the ability to start and stop virtual servers and services within minutes, and to 'scale' the capacity of each server as required, by adding or removing resources such as memory and storage space. While not a new configuration in itself[17], modern cloud computing has become the main computational infrastructure of Read/Write Internet by allowing development–centric organizations and individual developers to focus on their core expertise and products, clearly separating code *development* from code *execution* (and

[16] `http://www.opencompute.org/`.

the related material intricacies of managing the underlying infrastructure[18]).

Computation as a service and computational agency. As highlighted by scholars (Berry 2014; Jordan 2015; Mosco 2014) and activists (Stallman 2015), computation as a service is a disruptive reconfiguration of information politics and computational agency. On one hand, from the point of view of companies (from small startups to large corporations) providing services over the Internet to end users, the ability to outsource computation and to manage computational resources efficiently allows to focus on the company's core business without having to deal with financial, legal and information management issues stemming from having an internal team dedicated to managing the company's computational infrastructure. This is especially advantageous for startups, who can rely on computational capacity that can be easily adjusted to the organization's evolution. Critiques of computation as a service, however, highlight how the premium that is paid through the outsourcing of computational capacity is not only economic (computation as a service is normally more expensive than equivalent infrastructure that could be developed internally) but, most importantly, it involves surrendering control over computation: companies that rely on this can only control computation through the configuration options allowed by their service providers, and end users who rely on cloud services (for example, for personal storage or email services) cannot control how their data is processed and are often subject to very restrictive terms of service (Jordan 2015, pp91–92). Resorting to its characteristic strategy of creating new discursive spaces (Berry 2004), the Free Software Foundation introduced the label of 'Service as a Software Substitute' (SaaSS, with a reference to the widely known label of 'Software as a Service' or SaaS, which is one of the possible configurations of cloud computing), to highlight that by relying on computation as a service, users relinquish the ability to inspect, control and manage the software code and computational capacity used to provide services, in exchange of a black–boxed service whose computation is controlled by an untrusted third party. On the other hand,

[17] Computation as a service was in fact the norm in early history of computers, when users would submit computational tasks to central mainframe servers and wait for the results and outputs of their code to be returned to them.

[18] Such as making sure that enough data transfer and electrical power capacity are available, that the impact of hardware or network failures on user services is limited by moving affected virtual servers to different parts of the infrastructure, etc.

as will be discussed in Chapter 8, both privately owned companies and independent developer groups have been successfully developing FLOSS software code and related knowledge and best practices that allow competent users and organizations to create their own computation–as–a–service infrastructures, using hardware they own and control, effectively reproducing the advantages of cloud computing highlighted above without relinquishing control over computation.

4.5.1.2 Computational operations — DevOps

The ability to distribute computational tasks across cloud infrastructure can only be optimally exploited by automating the processes[19] required to do so: although complementary to the transition of computation to cloud infrastructure, these practices are analytically distinct and operate at different layers. Grouped under the label *DevOps*[20] (Debois 2009), these practices are focused on allowing to reliably configure, test, operate and monitor web applications and other Internet infrastructure in an automated way, by describing the desired state of a system[21] and the actions needed to bring it to the desired state, leaving to software code the task of keeping entire networks of servers in the state prescribed by the configuration, or changing states automatically to respond to events (for example, updating a web application's code to the latest version available). The core benefit of DevOps can therefore be considered the ability to replace traditional labour-intensive and potentially error–prone manual processes with testable and reproducible computational steps, which in turn allow to manage operations of large infrastructures at very low marginal cost compared to what was possible before the shift to computational operations.

Two main contexts for DevOps can be identified: on one hand, computational operations is necessary to operate reliable cloud infrastructures; on the other, cloud infrastruc-

[19]The term *orchestration* has emerged to indicate coordinated configuration and management of cloud computing infrastructures at a large scale.

[20]The term *DevOps* is a portmanteau derived from (software) *development* and (web infrastructure) *operations* and is generally used to indicate both the underlying operating mechanism (i.e. the primacy of code as enabler of operations), and the resulting configurations (i.e. that development and deployment of web applications code are closely related, so that developers can reliably perform updates to running web applications without long and intricate manual steps that require specialized systems administrators).

[21]Typically a server, although this can be any device: in fact the management of software and security upgrades on locked smartphones and tablets can be considered as computational operations oriented to consumer devices at very large scales.

ture can itself be more efficiently used as computational infrastructure by deploying and running web applications in an automated way through DevOps.

It important to note that the shift both towards cloud computing and towards DevOps is not as much technical as it is organizational (allocation of resources), epistemic (representation of strategies involved in the computational, large–scale configuration of infrastructures versus the manual methods previously used at smaller scales) and therefore ultimately political: whereas one of the core promises of these computation–based strategies is to make computation more easily accessible to anyone (this claim will be critiqued in detail in Chapter 7), larger organizations are those generally able to better leverage the computational efficiencies of cloud computing and DevOps, thereby further widening the gap in computational agency between those able to execute code and those unable to do so, or able to do so with less efficiency.

DevOps and computational agency. Developer discourses highlight how computational management of software infrastructure, similarly to reliance on cloud computing, has enabled smaller Internet organizations to compete with larger ones by avoiding inefficient resource allocation traditionally imposed by the lack of economies of scale: although some early Web 2.0 companies were already relying on DevOps practices (automation of management of code is discussed as competitive advantage in O'Reilly 2005b), these became systemic through the growth of the mainstream SNSes, whose operativity would be impossible without computational management of infrastructure[22], and of mobile operating systems (Google's Android and Apple's iOS), which intimately rely on the ability to reliably distribute, configure, manage and update software (apps) from remote, while needing to minimize any risks of leaving user devices in an unusable state due to malware attacks or misconfigurations. Alternative rationalizations discussed in Chapter 8, according to the developers accounts I have reviewed, are currently struggling to effectively employ DevOps strategies pervasively: on one hand, there seems to be wide consensus that only through computational automation of repetitive, complex and error–prone tasks independent projects can aim to provide reliable alternatives to mainstream Internet services; on the other hand, although DevOps software and best practices are well understood in scholarly research, their FLOSS implementations have been developing at a very fast pace in the last few years, making it difficult to rely on stable strategies that could support automation of software infrastructure with minimal expense of the normally scarce development resources available to independent projects.

4.5.1.3 Shift of web computational complexity to the browser

Whereas most early Internet users can easily relate to recollections of the pleasure of handcrafting web pages and associated assets such as images (Lialina 2014), in practice outside of the realm of early personal websites and handcrafted code-as-art most websites, from personal blogs to vast media/news sites and ecommerce sites, are routinely managed through specialized software systems (content management systems) which generate the web pages that users see by applying textual transformations on a set of source content through rules defined as software code. Most of the computational complexity of these software processes is managed on web application servers; each web page is generated and often stored on web servers, and this server-generated page is then sent to the browser of each visitor requesting it. Early Web 2.0 applications, on the other hand, started to routinely transfer part of this computational complexity to user browsers through the strategy later named 'Ajax'[23] (Garrett 2008), by running JavaScript code to request further data from web servers (possibly — and often — including servers other than the one from which they fetch the base HTML page) and by using it to generate elements of page content not present on the initial HTML page, without redirecting the user to a new page. Content fetched through the Ajax strategy, moreover, is often further transformed by applying calculations or context-specific rendering (for example, generating a chart from data in tabular format, by executing Javascript code within the user's browser).

Although the technical details of the Ajax strategy evolved significantly throughout the past decade, the basic principle has stayed the same: *alongside HTML content*, *software code* is sent to the browser, and this is then run by the browser and used to computationally augment the base HTML content. Initial motivations for the use of Ajax in Web 2.0 ap-

[22] It could be imagined that the pervasive automation of management of infrastructure could be replaced by a large number of engineers, although the economic and organizational implications of this would be difficult to assess.

[23] Asynchronous Javascript And XML.

plications included the ability to assemble web pages from content from different sources without the related increase of (server-side) page generation times, and while actually aiming to give to users the impression that updates to page content happen almost instantly as they would on a legacy desktop application reading data and content from the user's own computer.

This perceptual and aesthetic dimension of early web 2.0 apps, publicised by initial implementations of the Ajax strategy, contributed to originating the successful discourse of web 2.0 *applications* (as opposed to websites). From a technical point of view, web applications appear to the browser exactly as a 'traditional' website entirely generated on web servers would: as a series of resources sent by one or more web servers in response to a browser request triggered by user action. The key difference is that an increasing amount of the resources sent by web servers as part of a transaction set are software code, which then loads further content (including, possibly, further code, and so on).

Computation in the browser and computational agency. The overall progression towards shifting computational complexity to the browser (and, later on, to the operating system of smartphones and tablets, which in some cases is largely in itself a web browser, as in the case of the Firefox OS operating system) has served purposes beyond the initial ones outlined above. Besides context-specific implementations of Ajax-based Read/Write practices (letting users interact with visual affordances of the app/page to send content to remote services, updating the interface almost instantly to provide visual feedback), browser-based computation has increasingly been used, notably, to implement a wide set of user tracking strategies, ranging from gathering of data about app/website usage patterns to transparent and intrusive profiling aimed at fine-tuning the advertising displayed to users.

Accordingly, both architecture of web browsers and coding strategies evolved as well through progressive and interrelated changes: the JavaScript interpreters embedded in web browsers, which until around 2008 had been suitable only for basic user interface interactivity, have become very efficient computational environments able to run highly complex software code (Paul 2008; Pichai and Upson 2008) interfacing directly with the web as a platform for data interchange, on one hand, and with various sensors and features of user devices traditionally not available to be programmatically accessed from within browsers without the aid of plugins (such as GPS sensors, webcams, Bluetooth and NFC communication, access to local files, etc.) on the other, effectively making the browser an ubiquitous software platform able to connect users and their cognitive and physical environment to remote services, to data storage and to data processing capacity. An entirely new professional profile, that of 'frontend developer', emerged in parallel to the transformation of browsers from content rendering environments to computational platforms: as will be discussed in detail in Chapter 7, the development strategies developed in this new context and the learning paths for this professional profile constitute an epistemic change with significant implications for the ability for non-professional users to gain computational agency.

Whereas from the point of view of end users of lifeworld Internet services that exploit the browser as an efficient computational platform this shift of computational complexity to the browser is enabling user experiences that implement in practice what early Web 2.0 narratives promised (while actual implementations still needed to relegate most of the computational complexity to the backend servers, as browsers were not capable to sustain the kind of applications that can be run within them in recent times), developers involved in alternative rationalizations, as will be discussed in Chapter 8, highlight through their public discourses how the ability to rely on a pervasive platform (the Web) and on modern browsers as computational environments that can run the same web applications on different operating systems and underlying hardware enables to build applications that can be used and distributed widely without the complicated procedures that need to be followed in order to build and distribute 'native' applications across different operating systems, app stores, device types. Notably, the Mozilla foundation has been attempting to leverage the pervasiveness the Web as computational platform through an increasing focus on outreach and education initiatives aimed at fostering computational agency of users (both experts and common users, through different kinds of learning programmes), by helping them on one hand to understand basic (technical and policy) principles of the open web, and on the other to learn how to appropriate web applications.

4.5.1.4 Mobile general computation devices

Of the four aspects of the computational turn of the Internet discussed here and in later chapters as most relevant to the reconfiguration of user agency on the Read/Write Internet, the final one discussed here — the fast increasing uptake of mobile devices by end users — is the one more immediately visible to common end users themselves, and as such this aspect manifests itself not only in unprecedented levels of indirect and transparent infrastructural computational complexity but also through very tangible shifts in the cognitive representations of technical agency, through the reshaping of the digital boundaries of user lifeworlds and of the material practices of everyday user life.

Although global web traffic from mobile devices is only likely to reach that of desktop and laptop computers not earlier than 2016 (StatCounter 2015), smartphones and tablets have been gaining popularity among users in a very short timespan (roughly starting with the introduction of the first Apple iPhone in 2007). On a computational level, these devices are powerful general computation devices, employing a sophisticated software architecture to present functionality to users through installable applications, each of which typically provides a focused interface to specific computational capabilities of the device's processor, organised in a way to allow users to complete a well defined set of tasks through a touch-based screen interface.

Berry (2014, ch4) highlights how this touch interface contributes to hiding beneath the glass surface of the screen the largely intangible complexity of the software operating within devices (and, by extension, across the servers to which these are almost constantly connected); at the same time this interface also inscribes throughout the materiality of user practices specific patterns of interaction between the user's environment and the computational infrastructure providing services: namely, the absence by default of a physical keyboard at the same time 1. limits the amount of textual input that users can comfortably and rapidly provide, 2. shifts the users' attention to non-textual interaction patterns, such as taking photos or shooting video, manipulating on-screen lists of tags and other metadata, and 3. leads to reconfiguring the cognitive expectations of what is possible, practical and useful to do through these mobile interfaces, further contributing to a shift away from the textual and discursive dimension that characterized pre-Read/Write Internet, or more frequently to a reconfiguration of interaction patterns that relegate more textual and 'write-intensive' practices to laptop/desktop computers, while complementing these with 'read-intensive, write-light' and non-textual-oriented practices on mobile devices.

Mobile devices and computational agency. As for the other traits of the computational turn of the Internet discussed above, the ongoing transition to mobile devices also bears implications at the level of infrastructure and for developers. For example, the dominant mobile environments — those of Apple iOS and Google Android devices — rely extensively on cloud computing to provide services and to offload computational complexity from the actual devices for tasks that rely on low-latency analysis of big data; likewise, they depend on large-scale computational operations in order to manage software upgrades, app purchases and installation, as well as information security across hundred of millions of devices. Commercial — and, in the context of Read/Write Internet, ultimately political — considerations are also reflected in the tightly guarded permission schemes imposed on mobile devices: as monetization strategies for mobile platforms depend on Google and Apple's ability to dictate which software users can install, how they can use this software and which data can or must be provided by users throughout their use of their devices, the developers' toolkits for these platforms in practice bear the inscription of normative uses and of the overlaying of task-oriented interfaces to the devices' potential as general computation devices. Sufficiently determined users and developers may often be able to install software not distributed via the platform operators' official app stores, though in practice this often requires the use of arcane tools that can potentially leave the devices in a broken state if used incorrectly, further limiting the appeal of trying to use mobile devices as general computation devices to a small subset of committed users. My review of public discourses of developers involved in alternative rationalizations of lifeworld Internet highlighted how mobile devices are seen as an increasingly relevant terrain of struggle because of their pervasiveness and their often intimate role in each individual's everyday life — as also reported by some of my research participants while discussing how their smartphones would be accompanying most moments of their days. On one hand, the tight control operated through mobile platforms is seen as a problematic closure of spaces for computational agency, when users are left with little choice over what software to install on their devices, how to integrate features of mobile apps into their everyday lives, and how to deal with the management of personal data operated by app vendors and platform operators often more interested in gathering and analyzing user data than in the actual services and apps they are providing; accordingly, alternative versions of mobile operating systems (such as the many 'mods' — modifications — of the Android Open Source Project) that don't need to rely on proprietary Google services and on opaque cloud-based storage and management of personal data, as well as FLOSS alternatives to popular mobile apps are a growing area of focus for independent hackers, as will be discussed in Chapter 8.

4.5.2 An example: narratives of computational versions

As a final analytical endeavour of this chapter focused on reconfigurations of computation and of computational agency, I return to the topic of 'versions' discussed earlier on while deconstructing the *telos* of progress inscribed in mainstream Read/Write Internet affordances and practices, with the aim of briefly discussing through a practical and simple example the different considerations that are involved when dealing with the same topic of 'versions' within the domain of computational management of software infrastructure (DevOps trait discussed above).

If on one hand version numbers have come to stand for generic representations of stages of evolution in multiple contexts, in a format accessible to the general public (as articulated in the second section of this chapter), a material analysis of specific traits of the computational turn of the Internet can highlight how the meaning of versioning acts and signifiers has been transformed in recent years, contributing to reconfiguring machinic agency also thanks to the computational, machine-manageable counterpart of the versioning narratives discussed earlier. In recent years this computational configuration of versioning has been encoded in the widely adopted Semantic Versioning specification (Preston-Werner 2013), which constitutes an interesting example of how computational management of infrastructure needs to rely on a different set of engineering practices than those suitable to non-Internet software. Whereas, traditionally, software version numbers used to have mainly a marketing function and to give an approximate indication to developers and users of the age and maturity of an application, the Semantic Versioning specification is explicitly aimed at solving a software engineering problem that, although not new, became much more complicated throughout the shift towards development of large-scale web applications building on a multiplicity of heterogeneous libraries and APIs, often made available by different developers under a free software/open source license. Whereas highly centralised software engineering practices such as those typically used for the production of proprietary desktop software bring under the control of a single organization all the dependencies on which an application relies, the widespread practice of building web applications by reusing small, specialised libraries and APIs developed by third parties makes it indispensable to impose a strict control over the exact versions of any libraries and APIs on which a specific version of a web app depends.

The Semantic Versioning specification has quickly become the most widely used way to help developers formalise expectations: the core logic is stated in the specification's summary:

> Given a version number MAJOR.MINOR.PATCH, increment the:
> 1. MAJOR version when you make incompatible API changes,
> 2. MINOR version when you add functionality in a backwards-compatible manner, and
> 3. PATCH version when you make backwards-compatible bug fixes.
> (Preston-Werner 2013)

Although clearly useful when read by a human eye, the ultimate target of this formal description of change through versioning is a class of software applications essential to software engineering workflows of Read/Write Internet, known as *dependency managers*[24]: by instructing the dependency manager to use third party libraries at a specific version (or within a specific version range), through the use of simple text-based configuration files, a developer can make sure that any third-party library adhering to the Semantic Versioning specification on which their application relies is only used in a 'known good' version, and that the combination of main web application and its dependencies, each *locked* at a specific version (rather than using the most recent version available), can be extensively tested, therefore minimising the risk that incompatible changes in third-party code cause the web application to stop working or to work incorrectly.

Whereas the popular use of '2.0' and similar version labels contributed to creating narratives of linear progress and evolution, often with techno-optimistic undertones, the successive evolution and formalization of code versioning to be used by software dependency managers implicitly negates these linear narratives and acknowledges a more complex computational reality, whereby 'newer' is not necessarily 'better' (whatever this may mean in each specific context: more performant, providing more features, or offering a more polished user experience, etc.) nor 'desirable', and each change in the network of software interdependencies essential to delivering a web application to end users (but at the same time invisible to them) may introduce unexpected and undesirable behaviour, which can only be mitigated by computationally managing the stability of the network.

[24] E.g. Bundler for the Ruby programming language, NPM (Node Package Manager) for the JavaScript programming language, Composer for PHP, etc.

Chapter 5

Assembling lifeworld within Web 2.0: negotiations and normativity

5.1 Introduction

This chapter examines the case study of my own attempt (between 2005 and 2007) to develop a Web 2.0 application aimed at helping students of Goldsmiths university in London to critically reflect on their personal development within a digital information environment. Although genres such as blogs and wikis, as well as strategies to programmatically connect web services (through the use of APIs) and to exchange and aggregate information through standardized interfaces (e.g. RSS feeds) were already established at the time, how to meaningfully exploit the gleaned potential of Web 2.0 strategies and technologies in educational settings and within the individual private sphere (rather than to publicly contribute content to Web 2.0 platforms or to connect with others through early SNSes) was much less clear and open to experiment. Accordingly, the aim of the design of the app I developed ('3D Graduate') was to enable students to use any Web 2.0 apps they deemed relevant and useful, while providing an opportunity to aggregate content relevant to their personal development within a coherent interface.

Perhaps surprisingly, the relevance of this case study to the research topic of the dissertation is due to the fact that it ultimately failed: by framing it as an example of sociology of translation (Callon 1984), the failure to engage human and technical actors across the four key moments of problematisation, interessement, enrolment and mobilisation helps to disclose the diverging interests and forces that are here related to the wider systems of power that enabled some Web 2.0 configurations to gain mainstream relevance, while relegating alternatives to niche status. What at the time felt like a frustrating and shameful personal failure is here revisited to position it within the analytical framework of decisive computationalization of the architecture of lifeworld Internet. This stance allows to connect the practical issues discussed in the present chapter, and that led my project to ultimate failure, both backwards — to the hegemonic configurations of Read/Write Internet outlined in the previous chapter, and forward — to the alternative rationalizations discussed in chapters 7 and 8. The former configurations can then be understood, post factum, as the technical, social and cultural framing within which my project was situated, whereas the latter rationalizations have been seeking to address in recent years the power struggles that my project (as well as other similar ones at the time) could unlikely have acknowledged in the initial stages of Web 2.0 development.

My argument is that even within the nascent[25] narrative of Web 2.0, a complex and effective system of power relations was already taking shape. Some of these power relations were immediately visible and were promptly noticed and critiqued by scholars and activists[26], whereas other relations and their associated struggles, instead, were less visible by virtue of involving technical layers remote from the immediate user experience and at the time less familiar both to social researchers and to the technologists operating through them while developing Web 2.0 affordances.

Whereas this argument is illustrated through a case study limited in time and scope, in each section of the present chapter I examine how the context-specific tensions between

[25] At the time of the case study presented here.

[26] Notably: exploitation of free labour of users (Terranova 2004), erosion of privacy (Fuchs 2012; Sandoval 2012), growing control of private corporations and of governments over personal information and Internet practices (Jenkins 2008; Jordan 2015), as well as other issues outlined in Chapter 2.

social, technical and institutional domains could be reconnected to the issues surrounding the configuration of computational agency in lifeworld Internet throughout the hegemonic architecture and sub-genres of Web 2.0 affordances that were becoming popular at the time of the case study.

The chapter is divided into five sections. The *first section* outlines the original design of the 3D Graduate application as it was envisaged by myself and successfully 'pitched' at my job interview. The aim is to highlight how the emerging narrative of Web 2.0 information mashups was conjugated in practice to imagine an environment focused primarily on the lifeworld of individual students rather than on the institutional framing of personal development planning.

The *second section* examines the institutional context within which the project was situated, highlighting the tensions between my attempts to let students decide how to use the guidance for self reflection provided by the 3D Graduate application, on one side, and the institutional requirements related to dissemination of knowledge and to performance metrics, on the other side. The *third section* looks at the social and technical challenges that emerged through the development of the 3D Graduate application in a context in which general use of Web 2.0 applications and understanding of web information mashups was not significant enough to constitute an useful basis on which my proposed design could rely; the aim is to critique the complicated relationship between the delicate and unique peculiarities of user lifeworlds, and the technical form that was already being architecturally shaped to accommodate corporate 'walled gardens' (such as Facebook). The *fourth section* looks at the materiality of technical constraints within which the 3D Graduate application was being developed: the building blocks (tools and software libraries) and the software engineering best practices available at the time carried the inscription of specific and limited ways of organizing knowledge and of enabling social interactions mediated through software; the aim is to look at how these material inscriptions affected the development of Web 2.0 applications that wanted to deviate from nascent yet already hegemonic rationalizations.

In the *fifth section* I review the final shape of the project, analyzing the negotiations and the compromises that led to a much less ambitious technical form as well as to limited educational and social relevance of the project.

5.2 Design of the Goldsmiths 3D Graduate app: composing information mashups

My doctoral research project originally started as an action research project (see the Introduction for context and Chapter 3 for methodology notes), whereby I developed a web application as part of my day job at Goldsmiths (2005-2007) to support students' Personal Development Planning (PDP), structured around core Web 2.0 principles of interoperability between distinct web applications. I had intended to use this development project as my research case study, feeding back my research findings into further development of the PDP web application, but this plan eventually proved to be impractical as the university's management progressively steered the project towards a much simpler setup: rather than a web app, the PDP platform became a web-based repository of guidance documents and forms that students could download, print and fill in by hand.

Nevertheless, the research, design and development efforts involved in this project constitute an interesting overview of the materiality of Web 2.0 in a specific institutional context (an higher education institution) at a specific point in time (mid-2000s). The aim is to analyze the negotiations and configurations of computational agency through a case study of development of Read/Write Internet affordances characterized by a developer ethos privileging user freedom, an optimistic belief in the ability to support user agency of the open web protocols being developed at that time, and a belief in the possibility that technical and openness merits could lead to a widespread adoption of freely interoperable web apps versus the 'walled gardens' that effectively prevailed historically. At the same time, this critical deconstruction of my own Web 2.0 development project will highlight the limitations of my original design, when situated within the context of divergent agendas of different relevant social groups within the case study considered, and within the broader network of technical, economic and policy forces that were already influencing the dominant technical form of Web 2.0 and the spaces for user and developer agency.

The key traits of my design for a Personal Development Planning web app were substantially already defined in the presentation I gave to the recruitment panel at my job interview: in fact, in later discussions with my new colleagues who had sat on the job interview panel, I was told that the 'Web 2.0 approach' of my proposal contributed substantially to convincing the panel to offer the position of 'PDP Learning Technologist' to me.

Having met and gotten to know my new colleagues, I could retrospectively understand how my proposal could have captured their interest: the centre of which I became part, the Centre for Excellence in Learning Technologies, was a tiny group (four people besides myself) of talented learning technologists who were very knowledgeable about current best practices and curious about exploring experimental ways to improve the School's use of technologies to support teaching and learning: the profession of learning technologist was itself taking shape at the time, as most UK universities invested considerable resources in expertise that promised a high 'return on investment' by creating computational resources that could support teaching and learning and replace time-consuming and sometimes error-prone processes[27] with ones managed through specialised software, such as web-based course management systems; in my centre as in most other UK Higher Education institutions,[28] learning technologists were therefore essentially given the task of meaningfully integrating within the established learning and teaching practices of the institution the most promising Internet technologies of the time.

The core aim of my proposed design of the 3D Graduate web application was to help students manage information relevant to their personal development planning through tools and processes close to their everyday practices rather than forcing preconceived and normative knowledge representations and processes on them: thanks to this openness to individual needs and to the proposal to use Web 2.0 strategies it gained a sympathetic reception since the beginning between my technically minded colleagues.

Nevertheless, the overall design was rather simple: students would be encouraged to collect materials and information related to their personal development[29] through any

[27] E.g. submission and collection of course essays, distribution of course materials and readings, support to peer learning, etc.

[28] As I had a chance to experience in person during the development of my project whilst attending the yearly UK learning technology conference and being immersed in a rich programme of reports on often cutting-edge projects.

combination of the many Web 2.0 applications then available to manage different types of content (notable examples at the time were Flickr to collect photos, blogging platforms such as wordpress.com and blogger.com to store notes, diaries and other textual content, 43things.com for to-do lists), according to the kinds of content they found relevant and useful to collect, as well as to which Web 2.0 tools they found easy to use and, again, relevant to their specific requirements. The 3D Graduate web app would allow them to import the content they managed elsewhere, providing an overall view of all the materials accumulated through time and enabling each student to attach further notes to the content they would so aggregate, by establishing relationships between heterogeneous items (for example, linking portfolio items to development goals, development goals to timespans in a calendar, and so on). In my design, this was meant to offer ways to connect information *horizontally*, helping students to gain an overall understanding of the bits of their everyday achievements that they would consider relevant to their personal development.

Moreover, the web app was designed to provide ways to integrate what I represented as *vertical* connections: guidance, knowledge and guided processes to analyse aggregated content, provided by the university's central Learning and Teaching Office (of which my centre was a part) or by each student's academic department. This aspect of the web app more closely resembled examples I had seen while researching existing best practices of web-based PDP environments in other UK universities: however, whereas these were mainly sets of web forms modeled after an institutional view of knowledge and processes meant to provide expert guidance to students' PDP, my design started from the opposite end of the information spectrum: namely, personal content that was supposed to be closer to each student's interests and ways of thinking about information and personal development, on top of which a set of normative, institutional knowledge would function as guidance to reflect critically about goals and achievements.

Using typical Web 2.0 terminology, the architecture of my app was designed to enable *mashups* of content from multiple sources; differently from some similar web apps existing at that time (e.g. Yahoo! Pipes[30]), however, my app would explicitly split sources in two

[29] This included aims and objectives for their personal development, planning on how to achieve these, analysis of knowledge and resource gaps, materials (ranging from textual notes and handwritten sketches to audio or video recordings, as relevant to specific situations) meant as a record of achievements, etc.

layers: a *personal layer* (which, retrospectively, I would consider as constituting a lifeworld layer), assembled through criteria of relevance of items of content to each student's everyday life and personal development aims, and an *institutional layer*, composed mainly of guidelines on how to look at personal content within a personal development perspective.

Although it could not have been visible to me at the initial proposal and design stage, initial issues that would later intersect with others throughout this project were already taking shape at this point: at a moment that could be considered that of 'problematization' when employing ANT's framework of translation (Callon 1984), the definition of the problem to be addressed was not shared by all the actors involved. The institutional representation of what users (primarily students but also academic and support staff) should have been enabled to do through the web application was substantially different than mine: on one side — as will be examined in the following sections — there was an expectation that students should ultimately have been able to efficiently fill in some web forms, whereas my design relied significantly on the ability for students to autonomously develop their computational agency within a (web) space that provided only convenient ways to aggregate external content and some lightweight institutional guidance. The configuration of space for computational agency could be considered, ultimately, the obligatory passage point (ibid., p205) of the project's translation process, whose strength was undermined by the divergent implicit expectations between institution and myself as developer.

5.3 Institutional knowledge and information politics

Whereas the overall traits of my proposed design were substantially approved through my appointment, actually developing the application brought me in close contact with multiple forces of a complex information politics environment.

My work started by meeting PDP representatives from each university department in one-to-one meetings: these representatives were members of academic staff whose administrative and pastoral duties included overseeing the department's provision of PDP

[30] `http://pipes.yahoo.com/`. Somewhat ironically and sadly, this service is being discontinued by Yahoo! as I write up my dissertation, after just over eight years of public availability (Yahoo! 2015).

resources (knowledge, forms and guidance for self-assessment) to students, incorporating department-specific aspects of PDP. During these requirements analysis meetings, I would explain the project's goals and principles, and invite each departmental representative to introduce me to their existing PDP materials (these would often be forms accompanied by brief explanations) as well as to talk through any ideas they had about ways in which managing PDP via a web app could be enhanced meaningfully. As I would also discuss with each department representative a schematic representation of my proposed design, I was effectively attempting to establish the relevance of my web application as the tentative 'device' through which the requirements of each department and the spaces for student reflection could be computationally mediated, implicitly starting the phase of 'interessement' of the translation process:

> Interessement is the group of actions by which an entity [...] attempts to impose and stabilize the identity of the other actors it defines through its problematization. [...] To interest other actors is to build devices which can be placed between them and all other entities who want to define their identities otherwise. (Callon 1984, pp10–11)

Most of the department representatives expressed mixed feelings towards the planned web app during our meetings: the feedback I recorded included curiosity and expectation about the innovative architecture (especially between the colleagues who were somewhat familiar with the Web 2.0 hype of the time), some puzzled questions (mainly from the few colleagues who weren't active Internet users and who were not clear about which benefits a web-based PDP environment could bring) and widespread concerns about the possible effects on departmental administrative burden that an eventual success of the web app could have. This aspect of feedback turned out to be particularly relevant in terms of information politics surrounding my project: on one hand most departmental representatives were hoping — as did the university's senior management group — that shifting PDP provision to a web application could relieve the already stretched departmental administrative capacity from mundane tasks such as circulation of materials, running training and support sessions and answering students' questions; on the other hand, however, although most representatives were convinced that attention to personal development could meaningfully support and complement students' academic learning, they also expressed concerns about the ability for individual departments and for the central Learning and Teaching office to effectively support PDP and feared that by actually making it easier for students to engage in PDP activities, they would do so in increasingly larger numbers and that this would both strain departmental resources and possibly cause disappointment if students would feel left on their own without effective institutional support.

Substantially, it was implicitly made clear to me through multiple hints that departments were seeing my project as a senior management initiative that was welcome only as far as it would succeed both at improving support to students' PDP and at relieving administrative load related to PDP provision from departments; in practice, most departments took a 'wait and see' approach: they provided the key information and content I needed, but didn't offer to take an active role in the project, until they could understand whether the web app would be a net asset or liability to departmental resources.

Other key meetings that highlighted values and concerns that would be inscribed in the actual app design were with managers in the Learning and Teaching office; although I didn't meet top management staff, the middle managers I met relayed key requirements that were agreed directly with the Learning and Teaching vice-chancellor: these were, essentially, to make sure that there would be a common framework in place for each department and for central offices to publish PDP information and forms, for students to access these, and for data to be gathered about use of the web app, which was to be a key source for reports that senior managers would be asked to prepare regularly in order to confirm that the university was successfully complying with the relevant government body's[31] mandate about PDP provision to higher education students.

In this set of management meetings, the Web 2.0 architecture of the web app had a cold reception: through a mix of humorous and at time sarcastic remarks, it was commented upon as a facetious attempt to mix serious concerns (institutional knowledge and materials) with play and with pastime activities (personal content managed through Web 2.0 apps). My attempts to highlight that the intention was to closely bring together personal and institutional contents according to relevance to students' daily life in order to help make PDP part of individual lifeworld were unsuccessful. At this stage it was becoming

[31] HEA, The Higher Education Academy.

apparent to me that — even if the attempt to provide a web platform for PDP was genuinely meant to improve the students' learning experience — the institutional hierarchy of roles was being reflected in the way in which the extent of features expected from the 3D Graduate application: the full extent of computational agency that my plan aimed to promote was considered unhelpfully redundant in comparison with a simpler and more focused idea of agency implicit in the management group's plans. The Web 2.0 flair of my project started to be explicitly considered as a concession that had to be made to the 'geeky character' of myself and my co-workers involved in the project: as learning technologists (and therefore bearing the public image of somewhat radical web experimenters, as highlighted in the previous section) we could afford some reasonable freedom to experiment with latest web trends, as long as we would successfully develop the traits of the web app that could allow managers to report about the success of college-wide PDP initiatives. In practice, throughout this set of meetings the design of the web application was already being steered towards an architecture resembling the few existing publicly visible web PDP platforms at other universities at that time: repositories of forms and guided procedures that could be filled in and stored online.

Besides a fundamental divergence about the actual device around which the project would need to gravitate, this stage of planning also failed to effectively 'enrol' the actors in interdependent roles; most importantly, moreover, what was notably altogether missing at this stage was the voice of the core audience of the project: the university's students. I was, however, asked to start building the features of the web app that were considered essential by the senior management group before organising in-person meetings and focus groups with students and before creating questionnaires to gather student feedback; in line with management's view, the institutional knowledge components of the application didn't need any more research than what had already been done by departments and central offices while preparing the existing materials: these should be 'translated' to online versions first, and at that stage any Web 2.0 features could eventually be built.

5.4 User lifeworld and Web 2.0: a complicated relationship

Nevertheless, my research into ways to bring PDP in closer touch with students' everyday life and concerns, rather than adding it as a top-down institutional requirement to an already busy life/study balance, continued throughout the year of my project. My centre co-workers were very supportive of my approach and enthusiastic about sharing ideas and resources and providing feedback: like me, they considered the institutional requirements discussed in the previous section easily addressable with rather consolidated tools and were keen, instead, to help me build the traits of the web app that would, in our hopes, allow students to benefit from personal development planning. While building the Web 2.0 traits of the web app, several technical challenges had to be addressed: a brief analysis is outlined in this section, with reference to key Read/Write Internet issues discussed in the first part of the chapter.

Most Web 2.0 apps of the time allowed reuse of content in mashups by either providing a programmable interface (API) to query and fetch data or by making user data available as machine-processable data streams[32] besides showing it on web pages; in practice, however, most apps used unique API interfaces or data formats for exporting content, making it necessary to develop ad-hoc 'connectors' on the side of the apps consuming data (as was the case for my web application).

This posed a first, very tangible challenge: the range of external applications that my web app could interface with was inevitably going to be limited. In turn, this meant that I needed to prioritise specific applications, but I was lacking real use data amongst Goldsmiths students on which to base priority decisions, and I wasn't in the position to gather this data soon enough. A hard choice was therefore made to focus instead on what we identified as popular[33] applications providing rich sets of features (initially, Flickr and RSS feeds from blogging engines) and to promote these to students as external applications that they could use to manage content to be later imported into the 3D Graduate

[32] Such as RSS and Atom feeds.

web application.

It is interesting to note how even standard data interchange formats such as RSS and Atom feeds would in most cases need to be integrated on a case by case basis as almost all applications making data available for export in these formats either add custom extensions (as allowed by the relevant standards) or use RSS and Atom feeds simply as data transport channels, while the actual semantics of the data exchanged are dictated by the design of the app by which this is provided.

While trying to structure my web app according to the traits of interoperability of Web 2.0, addressing this functional requirement highlighted how, in practice, in order to allow users to connect any two web apps, substantial developer work was needed — both initially and ongoing, to ensure through time that any changes to APIs or data formats used by the web app exporting data would be correctly handled by the app consuming this data. In the following years, a niche market of web services specialized in providing curated 'integrations' between defined pairs of Read/Write applications has been developing, leading to the creation of services oriented to common users (for example, If This Then That — IFTTT[34]) or to enterprise workflows (for example, Zapier[35]); these services provide an intuitive interface that simplifies the underlying computational complexity inherent in coordinating disparate web applications through technical and corporate boundaries. The availability of APIs and data export and import formats is increasingly only a precondition to the ability to effectively computationally orchestrate flows of content, data and meaning over the Read/Write Internet: technical implementation details and the enclosure of Internet spaces within corporate silos (for example, major SNSes such as Google+ and Facebook) need to be taken into account when developing integrations. At the time of my development project, however, this kind of commodified interfaces was much less developed and could not rely yet on the knowledge accumulated in the following years nor on

[33]Not in the specific institutional context but on a global scale, as could be inferred from publicly disclosed usage data. Whereas the inability to properly assess use of Web 2.0 applications amongst students was due to organizational issues (i.e. failure to enrol students in the project through interviews and other methods aimed at understanding their needs), the reliance on data disclosed voluntarily by Web 2.0 companies (and therefore necessarily incomplete) highlights a systemic issue of proprietary web services within the domain of lifeworld Internet: whereas usage data is usually gathered and it could be argued that each user's data belongs to them — at least initially —, this is then often appropriated by the service operators for internal use only, effectively turning personal behaviour and attention data into analytics insights serving corporate interests rather than being made available to users to better understand their own practices online.

the computational capacity that makes the current implementations effective; early Web 2.0 applications would have to implement most of this integration software code independently: this represented a situated example of how the mainstream Web 2.0 discourse implicitly masks the necessary developer intervention behind a narrative of user freedom in connecting personal data stored in different applications: whereas the affordances (APIs or data interchange formats) may be in place, their use to convey user data between arbitrary couples of applications is well beyond the control of end users and their capacity to meaningfully connect data across applications. In the context of my app, this also meant that the initial design goal of letting students aggregate content managed in arbitrary apps most relevant to their needs and to their preferences in terms of user experience could only be met as far as resources to support arbitrary app integrations were available: beyond that, users would be invited to use specific apps explicitly supported by the 3D Graduate web app. The interface provided by these applications would still be richer than what could have been developed internally given the limited resources available, but relevance to individuals' daily life and interaction preferences could not match the expectations implicit in my initial design.

A further failure to effectively provide a meaningful computational environment to students became apparent in the difficulty to allow students to manage effectively non-digital content relevant to their personal development with as little *friction* as possible. According to research carried out by the Learning and Teaching Office before the start of my project, most students were dealing with large quantities of non-digital artifacts in their daily lives, and a sizable portion of students were relying on the Internet only for personal communication, for personal or coursework research and for purchasing goods and services, whereas only a minority of students were also curating personal content online.

In terms of materials involved in daily student life, whereas most coursework was already prepared and submitted in digital format at the time, even administrative tasks often involved printing out forms which would then be processed manually by departmental administrators. Moreover, as the University run a large visual arts programme, I was asked to

[34] `https://ifttt.com/`.

[35] `https://zapier.com/`.

take into account non-digital artifacts such as portfolio items that could be relevant to students' development while requiring to be digitised in order to be managed through the 3D Graduate web app. This posed additional challenges that required institutional collaboration beyond my own project and centre, and highlighted through material configurations (or more exactly, the difficulty to assemble these) further tensions and contradictions in the project's aims as formulated by senior management: large quantities of relevant artifacts needed specific processes and devices (scanners, digital still and video cameras) in order to be satisfactorily digitised and managed in digital format, yet no specific provision was incorporated in the management of the 3D Graduate project for any kind of digitisation.

My own attempts at organising at least a basic, pilot infrastructure that could be used to assess student use and eventually make the case for further resources, were not successful: after speaking with professionals involved in content digitisation at other universities, libraries and heritage institutions, I proposed to set up a small set of 'digitisation stations' in the University's library. These would allow students to scan documents or take photos of artworks produced as part of their course assignments and get their digitised content uploaded automatically to a personal storage area within their account of the 3D Graduate web app[36], where they could later link it to other content (for example, adding it to a web version of their portfolio). The institutional response to my proposal, predictably, was that this would have to be done through an entirely separate project, involving library and IT services resources, for which no budget nor other resources could be allocated within the short timespan of my project.

Interestingly, in a similar way to the challenge discussed above regarding the actual configurations of software required to interface web applications which already included affordances to exchange data but needed ad-hoc connectors, this attempt to bring non-digital content relevant to students within the digital realm of the 3D Graduate web app highlighted how most of the physical affordances were already in place (I had identified unused hardware such as older but perfectly suitable library computers, scanners and possible floor space for these 'digitisation stations') but the actual attainment of a functional

[36]To make the procedure as effortless as possible, my design included the possibility for students to identify themselves to the 'digitisation stations' simply by waving their student cards in front of a cheap barcode scanner such as those already in use in the library at book self check-out stations.

material configuration required the mobilisation of resources across different realms of institutional politics that in fact made the assembling of a viable solution practically infeasible. Similarly, digitisation devices such as digital cameras were already in the hands of several students (and most of them could either borrow a camera from friends or from the University's own media equipment centre); however this would hardly help to bring relevant content online as easily as my 'digitisation stations' idea would have: only a couple of years later, as smartphones and wifi-enabled cameras started becoming more widely available, taking photos and uploading them to some form of Internet storage became a simple practice not requiring much more than an initial setup[37] and actual shooting a photo, while uploading would be handled automatically by the smartphone or wifi-enabled camera.

The disconnect between institutional mandate and resource planning implicit in the difficulty to achieve a configuration of technology able to bridge the gap between non-digital content (relevant to students) and digital provision of PDP facilities (relevant to senior management's agenda) highlighted in a very material way what was already manifest in the words of managers I met at the beginning of my project: the attempt to encourage students to manage their personal development planning online was substantially modeled after not much more than a digital version of the printed materials that already existed, without a parallel remodelling of institutional expectations and practices and without an acknowledgement of the need to invest both in development of human capital (training, research), in infrastructure and in longer-term support to students, along the transition to a web-based environment.

The failure to address student needs in meaningful ways, however, was not limited to the ways in which Web 2.0 ideas were implemented in practice, through the thick network of actors and agendas involved, and taking into account technical constraints (analyzed more fully in the next section): my design, aimed at letting students integrating content already meaningful to them was an attempt to exploit the distinctive features of Web 2.0 applications, even before considering which content users would find meaningful to actually manage (and potentially share with others) within the digital domain of Web 2.0

[37] Often included in the download of an app and therefore accessible to common users without any need to obtain and enter obscure configuration settings.

applications. Reconsidering this project at distance, after almost a decade and through the critical framework developed throughout this dissertation, I would argue that my own design, although it was based on a genuine interest in the possibility to translate into technical materiality the complexity if individual lifeworlds, was essentially serving the purposes of a bureaucratic exercise that was forced on students, academic and support staff by the university's own management and by the higher education institutions that were setting requirements for universities to implement PDP opportunities for students, in a similar way to the bureaucratic and corporate takeover of online education analyzed by Hamilton and Feenberg (2012). However well-meaning and likely[38] supported by research and evidence these mandates may have been, they constituted an institutional technical rationalization that sought to impose rules and requirements based on an abstract generalization (Feenberg 2002, ch.7) of the complexity of individual life, while contradictorily hoping to create through Internet technologies a flexibility suitable to accommodate this complexity, without however questioning in the first place the relevance of the actual PDP process and of its (institutionally) desired outcomes to the specific condition of individual students. To illustrate this with the poignant image used by Berry of 'poorly designed website forms that we are increasingly required to fill in' (2014, pp39–40), my intent had been to design these 'forms' (more generally, Internet affordances) in a better way than most of the existing ones already employed in similar contexts, through empathic connection to students' interests and practices, whereas it was the very need for these 'forms' to exist that should have been critically questioned in the first place by all the actors involved. From an hacker perspective, the project would likely have appeared as being 'scratching a non-existent itch' (Coleman 2012a, ch.3).

5.5 Web 2.0 and development constraints

While designing, negotiating and building the 3D Graduate web app, issues related to the development of Read/Write Internet apps emerged as part of the ongoing reconfigura-

[38]Unfortunately the documents I collected during my action-research project don't include any substantial traces of such evidence, and neither do my notes or recollections of the materials I had access to whilst in my role in this project.

tions involved in the project. On a small scale, these issues, while specific to the context of my project, were indicative of the state of web tooling, knowledge and best practices of software engineering at the time, and as such they provide an useful insight over material constraints and values inscribed in Web 2.0 applications through their development, and — in turn, and ultimately — in spaces for users' computational agency.

A review of my archived email messages throughout the year of my project reveals interesting discussions with co-workers and external experts about development best practices typical of the time. One such aspect is the shared interest and attention towards standardisation efforts (as outlined in the first part of this chapter) concerning data exchange formats and ways to represent knowledge through protocols that could be processed by machines. Although an attention to formal protocols was widespread at the time, the discussions I had at the time were mainly sustained by the realisation that, as a tiny project with limited resources, our development efforts could benefit from consolidated protocols that were seen as carrying the inscription of knowledge about best practices developed through research efforts led by authoritative standardization bodies such as the World Wide Web Consortium (W3C) and the associated organizations and companies who were contributing to the development of *technical recommendations*.

However, as became apparent already at later stages of the project and even more so by analysing the relatively low and ad-hoc uptake of such standards across successful Web 2.0 apps, these imported protocols and knowledge models did actually stifle the development of my web app: in practice, an reliance on imported formalizations of knowledge introduced further negotiations that had to be made, besides those outlined in the preceding sections: in this case, I was trying to filter the already precarious balance of institutional, end user and developers representations through the putatively authoritative formal descriptions carried by relevant protocols; some of them, however (such as nascent protocols and APIs focused on web annotations, which are still in flux ten years later), proved to be a bad fit for our specific use case, while others (mainly those part of the W3C's *semantic web* efforts) incorporated ample provisions for extension, which however meant that only a general approach could be reused, while most of the actual formal structures would have to be developed anyways in a context-specific way, through our own research and formalisation.

Moreover, whereas negotiations with internal actors would happen through two-ways discussions, attempts to incorporate protocols and relevant standards proved to be a largely one-way effort: as formal documents sanctioned by standards bodies, and often further formalised in software libraries working as reference implementations, as a developer I could either accept and use them or reject them altogether and either try to implement relevant features myself or avoid providing them. A third way, made possible by the fact that the project relied exclusively on free software libraries, would have been to maintain private *forks*[39] of the relevant libraries, modified to suit project-specific needs: this, however, would have implied considerable development efforts through time, which were simply not possible given the limited scope and resources of my project. Given the rather limited sophistication of tooling and software engineering best practices available to individual developers at that time, the constraints implicit in software libraries developed by others, and into which specific computational rationalities had been inscribed proved to be not only opaque to end users (Berry 2014, pp39–40) but also to developers, who — despite having access to FLOSS software foundations on which to base derivative web applications — would nevertheless be unable, in practice, to exert any effective computational agency beyond the degrees of freedom explicitly inscribed in the software code reused.

A review of Web 2.0 apps that I made after the conclusion of my project interestingly highlighted how one of the key factors of distinction between apps that attained wide-scale adoption versus projects that remained largely academic proofs-of-concept was the way that external formalisations of knowledge were inscribed into software: whereas, for example, apps integrating semantic web principles and protocols developed mainly as niche products, most successful apps did instead build upon ad-hoc protocols developed internally to address their specific needs, avoiding costly negotiations of knowledge and translation against external formalisations. In the early years of Web 2.0 this was often manifest in the motto 'the simplest thing that could possibly work' (Venners 2004), and, in later startup discourse, expressed through the 'Minimum Viable Product' (MVP) label (and — often — cult, as discussed in Pittman 2015), as famously embraced by the early-years Face-

[39] Forks are versions of software projects developed in parallel to the original ones.

book through Mark Zuckerberg's motto 'Move fast and break things' (Zuckerberg 2011).

The tension between these two approaches, either privileging formal correctness of knowledge representations inscribed in software or pragmatically focusing on highly context-specific knowledge and practical solutions for its inscription in software, could be considered a purely gnoseological issue; from a material culture point of view, however, it is more useful to consider how sources of knowledge representations are translated in practice through the construction of artifacts (in this case, software and user experiences of the software). In the context of the 3D Graduate app, my attempts to commit to structures of knowledge representation produced by external experts made the aim of situating the user experience of the web app close to students' everyday concerns and lifeworld more difficult to achieve: a large amount of development time was actually spent trying to match expert knowledge to an initially superficial understanding of actual student needs, rather than investing a substantial portion of this time to research student needs in detail. Moreover, relying on imported structures of knowledge representation also implied embracing specific software engineering practices: as these theoretical structures were informing specific reference implementations, the actual set of development tools, data formats, software libraries and development best practices (what is collectively labelled as *tooling*) was substantially gravitating around the chosen formalizations. Whereas in later years 'lean' development practices (Poppendieck and Poppendieck 2003) became established, providing the material foundations for software engineering workflows that could be more easily informed by the understanding of the needs of each app's real end users, at the time of my project the available tooling sets were more consistently highly opinionated; in turn, this implied a practical tradeoff between reusing existing knowledge and tools to develop web applications quickly, relegating — on the other hand — user concerns to secondary importance, failing to incorporate insights from actual user lifeworld into the development of applications. Whereas large projects with ample resources could seek a better balance between these two aims, a tiny project like the one I developed would typically need to reuse as much established tooling as possible in order to meet project deadlines: in our context, this put an additional strain on the tension between development practices and user experience.

5.6 Engineering negotiations: actual configurations

Having discussed throughout the previous sections tangible instances of forces that shaped the development of the Goldsmiths 3D Graduate web application, I will outline in this final section the shape that the web application took at the end of slightly more than a year of development, negotiations and reconfigurations, highlighting how the combination of forces involved contributed to the final configuration.

Rather disappointingly, the product that was delivered and used for a few years[40], substantially unmodified, could certainly be classified as 'Web 2.0', although of a radically different shape than I had envisaged. The Goldsmiths 3D Graduate *platform* made available to students during the 2006-2007 academic year consisted of two main parts:

- a 3D Graduate microsite, hosted on the university's website and using the distinctive branding of the 3D Graduate programme; this website provided general guidance to students about personal development planning and listed PDF forms meant to be printed and filled in as paper documents; the content of this microsite was managed by the Learning and Teaching office, although some department-specific materials were provided by individual departments.

- a wiki platform, on which each student would get a personal, private space created at the beginning of their programme of study; this private space contained a set of web forms, adapted from the PDF ones published on the 3D Graduate microsite, that students could fill in and save online.

Notably, any real integration with external applications—my core design goal—was not present in any significant form: students could actually configure any number of external RSS or Atom sources to be displayed on one or more of their wiki space's web pages, but even when these would be connected to Web 2.0 apps on which students published personal content, the integration with the wiki component of the 3D Graduate platform was limited to displaying the latest few items in inverse chronological order, therefore provid-

[40] The wiki platform that was serving as document storage and editing environment has been unreachable since the end of 2010 (according to snapshots available at The Internet Archive), and the static web pages providing guidance documents and printable forms were deleted from the university's website, quite ironically, while I was writing these pages.

ing no more than an at-a-glance overview of the student's personal *Zeitgeist* throughout Web 2.0 apps[41], with no possibility to further contextualise and enrich the external data by connecting it to the student's personal development planning efforts.

A further third component of the 3D Graduate platform was perhaps the most interesting in terms of potential relevance to students' engagement with other Web 2.0 apps, but although it was made available to students towards the end of my project, it was never adopted as an official component of the institution-managed platform and it was decommissioned shortly after I moved on to other work projects. This component was an instance of the Elgg open source social network engine[42] hosted on the university's web infrastructure, and it provided some of the integration features of my original design. Similarly to the wiki space, integrations were possible only by using plain RSS and Atom feeds, but this specific tool allowed to reuse imported content to a certain extent, for example by adding notes to it or by sharing it with other users of the social network site based on the Elgg software.

At the end of a short period of 'beta testing' open to students, feedback was gathered through a short questionnaire before the new platform's official launch; although not representative of overall trends due to the low response rate (less than 40 responses were received), it was interesting to note that student feedback on the more advanced features of the platform (Web 2.0 integrations) was substantially polarised: on one hand a few students were enthusiastic of what they saw as a commitment of the university to innovative ways to use web applications, whereas another group of students stated that they were unclear about what could be achieved through the Web 2.0 traits of the platform. Nevertheless, the Learning and Teaching office considered the basic requirements to be met by the combination of informational microsite and wiki space, and decided that due to lack of resources the social network component of the platform was to be considered an experiment and would not be kept active.

In late conversations with key decision makers, it emerged that part of the decision

[41] Moreover, only content made openly available on the external Web 2.0 applications could be imported in each student's wiki space, with obvious privacy implications, due to the often private nature of content that students were expected to aggregate.

[42] `https://elgg.org/`.

not to invest in the social network part of the platform was also shaped by the essentially dismissive opinions that they were forming at the time about the social network (Facebook) which was becoming very popular amongst students at the time, which they considered at that stage essentially a space where students would just hang out with their friends, rather than an increasingly important part of students' everyday life, used both for play and for serious purposes.

5.7 Conclusion

As most of the colleagues involved in some form in the 3D Graduate platform took on roles at other institutions in the months following my shift to other work projects, I have been unable to gain access to data regarding use of the platform in the successive years, or to student and institutional feedback past my departure. It is, however, interesting to summarise how a complex network of forces contributed to shaping what was born and ended up being delivered as a project fully imbued in the Web 2.0 climate of the time, although taking a very different shape along its development.

- Although the 3D Graduate app was targeted at students, institutional stakeholders nevertheless saw themselves as important *users* of the app: either as providers of content (departmental representatives) or as users in charge of overseeing, monitoring and reporting on platform use (management); accordingly, their requests, informed by substantially different representations of what the platform should like like than those of students, shaped the development of the project in ways often contrasting with the stakeholders' own assessment of what students needed.

- As institutional stakeholders had no thorough understanding of the potential to interconnect heterogeneous data through Web 2.0 apps, their expectations related to a web app were shaped by the existing 'offline' provision of PDP resources; accordingly, the institutional framework used to assess the success of web PDP initiatives was similarly using existing resources as reference: this process of remediation (Bolter and Grusin 1999) therefore not only skewed substantially the development efforts away from the more radical Web 2.0 traits, but also 'fudged' (Slater and Aritzía 2009) any ongoing and final assessments of the development project, as any web app feature was improperly compared to paper-based provision.

- Student input was severely impaired and couldn't contribute in any significant way to the shaping of the application: institutional requirements and priorities postponed student engagement to the stage of final evaluation, therefore precluding any real possibility of taking their feedback into account, except for the anecdotal evidence gathered by individual departments without using any shared methodology allowing for meaningful comparisons between contexts.

- The attempts to make the web app relevant to students' lifeworld incurred technical challenges as well: the narrative of interconnection between Web 2.0 apps needs to be contrasted with the reality that ad-hoc integrations almost always need to be developed between any pair of web apps before users are able to meaningfully connect apps. Moreover, although the Internet was already a part of students' everyday lives at the time of the 3D Graduate project, it was still difficult to bridge the gap between 'online' and offline—a disconnect which has largely become less relevant in more recent years thanks to Internet–connected portable devices that allow low–friction sharing of photos, audio and video to happen.

By presenting this small scale but nevertheless intricate case study of the development of a web app at the height of the Web 2.0 excitement just after 2005, my intention has been to analyse how multifarious factors, often not technical at all, can interact in shaping the design and development of web applications, progressively inscribing heterogeneous values and agendas in software. Even when the overall goal is to provide ways for end users to use web apps as Read/Write, the extent to which writing is allowed and the specific conditions and processes under which it can happen end up, accordingly, being part of the materiality of the software artifacts that are made available to users.

Chapter 6

Re-assembling lifeworld Internet: user accounts

6.1 Introduction

This chapter examines how common users reassemble the Internet in their everyday life, reconfiguring and appropriating practices, narratives and technical affordances as part of daily activities and concerns, and according to personal needs, priorities, strategies and aesthetic considerations.

The aim of the chapter is to explore the material extent and shape of what is constituted as 'lifeworld Internet' throughout the actual practices of individuals, and how individuals' computational agency is configured within this domain of lifeworld Internet. The timing of the fieldwork on which this chapter is based (2010-2011) is particularly relevant to the dissertation overall focus on agency through the *computational turn* of the Internet — a precipitous transition to a much higher proportional reliance of most Internet activity on computation than at any previous time (Chapter 4): my return to the site of the first stage of fieldwork (analyzed in the previous chapter) coincided with the key timeframe of this transition. Although its character and extent were not yet clear to me at that time[43], when analyzing the fieldwork interviews later on I could connect changes in user attitudes and practices — as discussed by my research participants — to the ongoing computational turn, as will be discussed here.

This chapter is based on the following three interwoven strands of analysis, starting from the materiality of everyday life: the aim is to explore the colonization of Internet technologies by everyday life, rather than the complementary process of colonization of everyday life by Internet technologies (certainly interesting in its own right, and widely explored elsewhere: see Greenfield 2006).

- Users' accounts of what constitute everyday concerns and of the traits of everyday life for which a technological mediation is felt relevant and useful: the *lifeworld horizon* of each user is the starting point of analysis, rather than trying to analyze how genres of Internet apps 'popular' at the time were incorporated in everyday life; what I observed was that strategies and practices of appropriation employed by each user were intimately connected with personal convictions and ways of approaching life situations not directly connected with the technical domain.

- The actual *practices* involved in the interpretation and appropriation of available Internet affordances, aimed at meaningfully matching individual representations; whereas this strand of analysis also includes an overview of the technical affordances involved, it primarily highlighted a wide array of social practices beyond the immediate technical domain that were an essential part of the actual configuration of lifeworld Internet (for example, enlisting a relative for support in order to successfully appropriate an unfamiliar affordance).

- Most importantly — within the dissertation's core research question — how *user*

[43] Like others at the time, however, I had an intuitive understanding of an ongoing transition happening, although I was not able to frame it as clearly as further years of reflection and perusing of scholarly literature would allow me to do. A short note from my private diary dated 30 December 2010 — in the middle of my 2010-2011 fieldwork — lists a few epiphenomena which captured key traits of the transition in process: most of these relate to the improving capacity to efficiently develop and run code on the Web, although the ability to reconduce them to an ongoing *computational turn* was missing: *'2010 has been a great year for web developers using free software and modern standards, with major developments for frameworks (Catalyst and all the Modern Perl stuff, PSGI/Plack, a new major release of Rails), [...] JavaScript becoming more and more usable, and—among many other exciting developments—HTML5 and CSS3 coming together at a sustained pace and usable across most of the major modern browsers. [...] [C]omplex infrastructures are now easier to design, deploy and maintain thanks to mature virtualization and orchestration software. It's great to see how the web is becoming so central in software development—and it's exciting to imagine how these tools could become even easier to use in the coming year, allowing easier connection between stuff and people.'*

> *agency* is configured, through the negotiations between available technical form and desired outcomes: whereas most of the Internet apps and services discussed by the fieldwork participants included some *designed* degree of configurability, the ability of each user to meaningfully appropriate them within their everyday life depended on factors such as technical expertise, availability of alternatives amongst which a user can choose, ability to match a personal representation of features of specific affordances to actual individual needs.

Everyday life is therefore the starting point of an exploration that extends to the technical affordances available to users and the underlying infrastructure: exploring everyday life practices without paying due attention to the increasing expansion of technical rationality that inscribes identity thinking (Berry 2014, p12) in users' lifeworld would deprive the analysis of the essential grounding in the issues of power that inevitably constrain user agency.

My argument is that user agency within the domain of lifeworld Internet can be meaningfully explored through the analysis of the role of computation as the site of mediation between the social and the technical, and through the interpretive framework of *computational agency* (introduced in Chapter 4) which subsumes the capacity to act within computational structures and the recursive capacity to reconfigure these computational structures to various degrees.

Whereas the previous chapter explored the power relations already operating within the nascent narratives of user choice of Web 2.0, the present chapter's role within the structure of the dissertation is to develop an understanding of user practices and agency at the pivotal time of the computational turn of the Internet. This understanding is then employed to empirically ground the argument — developed in Chapters 7 and 8 — that the improved capacity to exploit computational capacity so far seized mainly by large corporations intent in centralizing control over the technical infrastructure of the Internet can also be potentially used by independent groups of hackers as the technical foundation on which alternative rationalizations can be built, informed by a closer understanding of actual needs and practices of end users.

This chapter is composed of three sections, each of which looks at a distinct trait of computational agency within the domain of lifeworld Internet. In the *first section* I analyze whether the wider availability of technical affordances that allow to exploit the Internet as a Read/Write medium affected users' agency by actually empowering them to reconfigure their lifeworld Internet. Within this case study, this is done by assessing to what extent, and how, the original Web 2.0 promises related to the ability to easily create and share content and to configure personalized Internet environments thanks to the interoperability of web applications — both assumptions on which my Web 2.0 project (Chapter 5) relied — had actually translated into an increased capacity for common users to integrate disparate aspects of their Internet-connected lives. In the *second section* I look at computational agency through the tension between norms and actual practices as discussed by students in relation to their use of the Internet: specifically, two of the themes that consistently emerged across student accounts were the changes in perception and practices related to downloading and sharing content and the strategies employed to deal with social norms online. In the *third section* I focus on how computational agency was being reconfigured by and around users at the time of my fieldwork; firstly, on how users were making sense of the growing array of applications and devices available to them and of their ability to configure a computational environment meaningful to them; secondly, I look at deeper relationships with code and computation through the accounts of some students who disclosed some significant degree of technical expertise and discussed at length the benefits they perceived from the ability to deeply re-assemble their Internet experience thanks to the possibility to study, 'hack' and appropriate free/open source software, beyond the reconfigurations operated by other students who were simply choosing and combining, without complex modifications, available affordances.

6.2 Five years of Web 2.0 promises: what really happened?

6.2.1 Producing content on the Read/Write Internet

As discussed in the previous chapter, one of the factors that had contributed to the failure of my Web 2.0 development project was the low uptake of the more sophisticated traits of Read/Write Internet practices amongst the students for whom my project had been designed. Therefore, as I set out to return to the same fieldwork site almost five years later, one of my central concerns was to understand through my fieldwork interviews how far the Web 2.0 promises of user choice and Read/Write use of the Internet had been incorporated by students in their daily lives during the intervening years. The rows upon rows of computer screens displaying the unmistakable blue-white Facebook interface throughout the campus library — where I was spending most of my time inbetween fieldwork interviews — made it clear that at least Facebook's orchestrated forms of Read/Write interactions through the Internet were exceedingly popular amongst students at this new stage.

When discussing actual practices and motivations with students, a wide range of strategies emerged, highlighting nuanced approaches to production and sharing of digital content. With the exception of a few students (whose stories are discussed in the last section of this chapter), most participants relied on 'SaaS' web applications, that they therefore used essentially as a cloud service, without direct access to the underlying software code and data — including their own, besides what allowed by the apps' user interface. Nevertheless, none of these students expressed substantial concerns or disappointment for the lack of direct control over the software used: their accounts focused instead on the extent to which they were able to meaningfully support their daily concerns through the software affordances available to them. The perceived centrality of appropriateness of software affordances and of their uses therefore complicates the actual configuration of computational agency: for these students, computational agency did not only depend on technical competence and on the malleability of the affordances that they encountered, but was also negotiated against pre-existing cultural representations (such as those articulated as remediation in Bolter and Grusin 1999), was progressively acquired through the ability to experiment (whose monetary and emotional cost can be much lower when dealing with digital strategies as compared to non-digital ones), and was dependent on context: users who are knowledgeable and have experience of reconfiguring their Internet environment may choose to limit their practices to simpler forms if these provide a better fit to specific use contexts.

6.2.1.1 Remediation

Whereas most of the students interviewed did make a Read/Write use of the Internet, to various extent, almost none of them saw themselves as 'content producers': although they were creating and publishing content, in some cases in large quantities, as this happened mainly on social network sites the social aspect of their practice was considered by themselves as the actual context. For some students, a distinct split between professional and non-professional content producers still informed their perception of whether their own practices should be considered as 'content production': although this split has been considered by some (e.g. Keen 2007) as increasingly blurring within public perception, expectations that could be explained through Bolter and Grusin's (1999) concept of *remediation* kept it relevant in students' accounts even if their own practices were often situated somewhere inbetween the distinct ends of the spectrum. Kathy, for example, saw her public travel blog, to which she had been adding content most weekends from an Internet café in Kenya, as a personal endeavour 'just for family' and essentially as an edited version of a handwritten diary that she was updating daily, whose public status was only incidental and due to the fact that she wanted it to be accessible by her family members without forcing them to create accounts on the travel writing website she was using for her own blog.

The choice of tools used to publish content online was also seen as an important factor of distinction between casual and professional production; accordingly, different levels of technical expertise were considered necessary within the two contexts: although Kathy didn't consider her lack of deep technical expertise to be a limiting factor to her everyday Internet practices, she chose to join a travel-focused blogging site that limited her ability to customize the visual layout of her travel blog but allowed her to just focus on writing her occasional, semi-private posts; Hye-jung, instead, clearly saw her own travel blog posts as a professional endeavour and accordingly had been hosting and maintaining her own WordPress blog, learning how to deal with code and with software updates and how to tweak the visual theme of her site in order to provide an impression of careful curation of her travel diaries even from an aesthetic point of view. Although Kathy stated that she would consider trying to do travel writing for a living later on, she didn't feel that her first public blogging experience, although it ended up being read by several strangers besides her own family, had invested her with a professional role, whereas Hye-jung's self perception had been built through time as she accumulated subscribers to her blog and as she obtained an informal sanctioning by publishing her first printed travel guide: even while carefully curating her travel blog, she still felt that her online writing was a sort of 'rough cut' version of what she would then meticulously edit for printed guides — the ultimately 'professional' product of her creative work.

These accounts highlight how the definition of content production, as well as that of content itself, was spread across a broad spectrum for my fieldwork participants, as opposed to a more dualistic view embedded in pre-read/write Internet discourse. In a way, this complicates the trope that on today's Internet, everyone can publish content: technical ability is only one side of a complex practice, which also includes motivation (or lack of it, such as in Ian's case), availability of time (Emir), perception of one's public image (Kathy and Hye-jung), and most importantly context, as discussed below.

6.2.1.2 Making sense of the ability to produce content

Some students who had been producing content to be shown or performed in public reverted to a more private attitude because of life situations: Rosa was taking a year off her work career to study Visual Anthropology; whereas her previous documentary video work had been published online before, during her Masters year she was focusing on production of new video content as related to her coursework: accordingly, she stated that she wasn't making new videos available to a wider public, but just posting them online to gather feedback from a close friend living in her home city, with whom she exchanged opinions on their respective works-in-progress.

Similarly, Rachel had been working for several years as a music composer for a mainstream music label; once she resigned because of disappointment with the work conditions, she started composing music without it being commissioned commercially, initially for her own enjoyment and to develop a new creative stream, publishing and promoting it online with the main aim of getting acquainted with the many ways in which she could promote her own music through the Internet trying to make her creative work her main source of income again, without the commercial pressures of her former music label. Although clearly of professional quality, her music became more experimental both in aesthetic form and through the ways in which Rachel promoted it online: she stated that, at the time, she was trying to build an audience for the music she posted on SoundCloud (a public website aimed at allowing users to share their own music and sound recordings), with the help of social media (Facebook). She also experimented with using YouTube, for a different side of her creative output: she posted there videos of her 'stealth piano' project — informal videos, unedited or only slightly edited, of herself 'breaking into' rooms in public buildings with unused pianos in them, playing cover versions of popular tunes. She also kept a blog, sharing there both her music and her thoughts about 'going solo' after a career in the mainstream media industry, although she claimed that she wasn't convinced of the usefulness of this and that she was mainly experimenting with it having been pressured by a friend who was trying a similar 'going solo' route at the same time: having detached herself from what she described as 'factory-style' micromanagement within the music cultural industry, she was attempting to make sense of the regained freedom by experimenting with suggestions from friends and from online discussions in order to find a balance of strategies that could work for her and according to her personal style of public artistic engagement. In turn, she also reported that her style of engagement was being reshaped by her experiences of direct access to audiences, which had made her more sensitive to the nuances of each strategy tried and to their combination, in a different way but with similar material grounding to what Madianou and Miller recently described as polymedia (2012; 2013).

Context was therefore seen as a further distinction trait of what counts as 'content' worthy of being published and shared widely. Both Rosa and Rachel, albeit professional content creators, throughout their year of postgraduate study focused on private sharing of artistic output, with the aim, respectively, of publishing work on the Internet to gather private feedback and of getting acquainted with production and distribution of digital content without intermediaries. The actual content produced here was not considered important per se: the aim was not 'to be heard', which in popular discourse is seen as one of the revolutionary potentials of the Read/Write Internet, simplistically omitting that in a context where millions of people all try to be heard, the resulting effect is that most of the published content is largely relegated to obscurity and hardly ever accessed by a wider public. Neither was Rosa's or Rachel's main concern to express an artistic or political message through their publicly accessible work: this had been at the forefront of their professional activity in the past, but at this stage both students were mainly trying to *make sense* of what the ability to easily create and publish content could mean *in their context*, which of the many Read/Write affordances of the Internet were most relevant and meaningful for their own future activity, and how to best use these in order to gain broader visibility for their future research and artistic output.

6.2.2 Aggregating content, connecting meaning

As one of the defining traits of the design of the web application that I had been developing for Goldsmiths was the ability to integrate heterogeneous streams of content from arbitrary Web 2.0 applications, I was interested in assessing through my fieldwork whether and how content mashups would actually form part of users' everyday Internet practices. This question has a particular importance in the context of my research, as my working hypothesis about integration of content from different web applications was that interoperability between arbitrary applications would empower ordinary users, by allowing them to publish personal data and content on any web application they felt comfortable using, while still being able to create coherent presentations of their content by 'pulling in' different bits from distinct, yet interoperable, web applications: although one of the distinctive traits of Web 2.0 applications in public discourse is that they can enable any Internet user, regardless of their technical expertise, to publish content easily, my focus here is on specifically on the computational mechanics of how distinct sites and services are integrated within a coherent (from the students' point of view) experience.

When considering users' practices literally, every research participant reported using web apps that provided the ability to 'aggregate' content from different sources according to users' intentions: most typically, even for students like Ian and Emir, who maintained that they were making a very limited and somewhat basic use of the Internet at the time, this meant at least curating one's profile and sharing some content on a social network site; every student was using Facebook to some extent, and some students were also active on other social networks.

Facebook and other major SNSes provide an highly curated user experience, aimed at making it as immediate and as easy as possible for users to contribute and aggregate content (photos, status updates, blog post-like texts, links to videos and to content elsewhere). From the SNS operators' point of view this kind of user experience — often referred to as 'frictionless sharing' — is meticulously and continuously optimised[44] in order to maximise users' contribution and curation of content, on whose growth the operator's business aims often depend; from the user's point of view, instead, this translates into a reduced reliance on intentionality when sharing content and into a simplification of the publishing process when creating content.

Through my interviews, instead, I explicitly explored how users *intentionally* choose how to assemble content; similarly to the complex configurations of computational agency related to production of content discussed in the previous section, the fieldwork findings outlined a rather nuanced picture. Although basically no real 'Web 2.0-style' content mashup practice was described by any of the participants, most of them did actually make use of web applications chosen because of the degree of control (workflow, visual presentation of content, choice over who to share content with) offered to users, as opposed to the vendor-curated user experience of SNSes. Social network sites were nevertheless widely used as well to publish content; a rather clear separation was often observed between content posted on SNSes and on specialised web apps, both in terms of type of content (e.g. personal thoughts or everyday banter versus content aimed at a wider public) and in how this was presented.

The most widely used types of Web 2.0 apps were blogs (different platforms were used: WordPress, Blogspot and other, specialised ones such as travellerspoint.com, as well as a few self-hosted WordPress blogs) and photo sharing websites (mostly Flickr); each student who was writing a blog connected this to a specific reason and context, as opposed to the occasional, often unplanned posting of text and photos on Facebook: Peter had been blogging throughout his long bicycle journey from the UK to India; Hye-jung had been regularly mixing personal thoughts and informal updates from her journeys on her per-

[44]Cfr. trait 4 — End of the Software Release Cycle in O'Reilly's (2005b) Web 2.0 definition.

sonal blog since 2001; Kathy had been updating a travel blog in the few weeks she had spent doing volunteer work in Kenya during her gap year: as communicating with everyone in her large family was difficult and expensive during her time abroad, posting short updates every weekend was a simple way to keep all her relatives reassured that she was well and was enjoying her experience in Africa; Markus had created several self-hosted WordPress blogs for different projects he had been running recently; Rachel started blogging about her own experience of starting a solo music career after having been employed as a composer by a media company.

Interestingly, when looking at the blogs of these students, it appeared that all of them were indeed integrating content from different web apps to some extent, although this typically was in a rather basic way and often according to options offered by the blogging platform used (e.g. photo widgets displaying the most recent photos from their Flickr account on the sidebar of the blog, links to their other public accounts on the Internet, such as LinkedIn or Twitter) rather than by employing more customized and unique mashups as those often showcased in Web 2.0 advocacy blog posts — for example, mashups built through Yahoo! Pipes, a web app which allowed users to connect distinct web apps through a visual interface.

Some users actively mixed content from different web apps, although this was done manually, rather through the computational procedures that had been envisioned as the engine of Web 2.0-style mashups. Peter and Rachel manually cross-posted new content across their Facebook and Twitter accounts and their blogs, others manually embedded photos from their Flickr accounts on their blog posts. This, however, was basically the apex of content integration practices as observed through my fieldwork: the computational integration seen as distinctive in the Web 2.0 narrative was not something that ordinary users, at least within the group of my fieldwork participants, would engage in. Conversely, every student interviewed was bringing together different kinds of content (text, photos, videos, music) *within* the standardized Facebook user experience, which is explicitly designed to be used by ordinary users, hiding the computational complexity which was instead a distinctive trait of web apps such as Yahoo! Pipes, visible through the 'geeky' feel of the use cases through which these Web 2.0 apps were marketed.

The failure of the user-empowering flair of the original Web 2.0 vision could be considered to follow partly from the lack of tools allowing ordinary users to integrate content from different sources as easily as they could do this within curated environments such as those of Facebook and other SNSes, partly from the fact that most users seem to prefer rather simple ways of presenting their content on the Internet rather than expressing a need for more articulate integrations, and partly from the incredible success of platforms such as Facebook in capturing users' attention and in providing an intuitive interface for frictionless sharing of content: how each of these causes concurred to the prevalence of curated SNS environments over the orchestration of independent apps part of the Web 2.0 vision would be hard to assess other than through large scale investigations, also considering that different factors may make one of these two content integration strategies (computational versus manually curated) more or less appealing to each individual user. However, what the accounts of my fieldwork participants highlighted is that they were much less concerned with the technicalities of how to bring together distinct sets of content that is managed through different applications for practical reasons (e.g. Flickr was considered, by students using it, as providing a higher quality user experience for sharing photos than Facebook photo albums) than they were with finding a way, through the tools readily available to them, to present content according to their precise needs, again highlighting how they would see the Internet as an integral part of their lifeworld, and structure their choice and use of tools accordingly.

6.3 Reconfiguring practices

Whereas the conversations that I had had with students at the time of my Web 2.0 development project often highlighted a mix of curiosity for the possible uses of the Internet and of uncertainty about what they could actually do online, during this second phase of fieldwork almost every student showed fuller awareness of what they could expect from Internet applications. What students discussed at length, and reported as most relevant to them, was not the technical aspects of their practices but how existing practices and norms would need to be reconfigured when dealing with them through the Internet. As for the traits outlined in the previous section, most students were satisfied with their level of technical expertise and were instead actively experimenting how to match the technical form to their expectations: the following section examines how the *mechanics* of downloading content via filesharing networks were familiar to all the students interviewed, whereas the *implications* of the availability of large quantities of content were still being explored (e.g. how to meaningfully consume and share content).

6.3.1 Downloading, consuming and sharing content

One of the distinctive examples of Web 2.0 technologies listed in O'Reilly's seminal article (O'Reilly 2005b) is BitTorrent, one of the most widely used peer-to-peer content distribution systems. O'Reilly's article contrasts BitTorrent, as a Web 2.0 technology based on an inherently Read/Write architecture (whereby each user is simultaneously downloading and uploading content), to Akamai (which at the time was a one-way server-to-client content delivery network) as a pre-Web 2.0 technology, focusing on the efficiency and related savings that Web 2.0 companies could achieve by offloading part of their content delivery infrastructure to their own users. In the following years, however, BitTorrent effectively became more widely successful as one of the main ways for users to gain access to cultural artifacts such as music tracks and movies. The BitTorrent protocol is indeed used by many organizations to distribute content, but this client-to-server use — the one anticipated by O'Reilly — has remained largely subordinate to its role as infrastructure for unregulated filesharing between individuals.

For most of my fieldwork participants, p2p filesharing had become the norm in terms of procuring content, although in their accounts its computationally orchestrated Read/Write architecture[45] appeared to have been largely black-boxed behind the convenience of being able to quickly and easily download most digital cultural artifacts they were interested in. Conversely, forms of intentionally sharing the *consumption* of content were discussed by some fieldwork participants as highly relevant to their everyday life and as part of the configurations through which they made sense of their Internet use.

[45] For example, the negotiation of upload and download ratios between any given pair of peers, the background curation of network routes to peers in a 'swarm' to improve discoverability of peers who possess specific chunks of content, etc.

The preferred way to obtain music and movies as described by most fieldwork participants was through 'downloading' files: while *downloading*, technically, just means transferring content from some Internet source to one's own computer, in practice it was universally used by the students to indicate that they were obtaining content through filesharing networks. This was considered the *de facto* normal way of gaining access to content: some students also purchased music tracks via iTunes or other websites operating in agreement with the media industries, but mostly on specific circumstances.

Kathy discussed her filesharing habits in detail, stating that she downloaded '**quite a bit, actually [chuckles]**', and that she would only pay for music as an exception:

> **Sometimes if I'm really desperate I'll buy it, but... I don't really like buying these things, it's really really bad, but... [chuckles] [...] The other morning I was walking to uni, it was really cold and I was in a really miserable mood and I wanted a really happy song so I just thought, you know, I'm just gonna download this on my phone right now, just pay on iTunes and do that.**

Ease of access and economic convenience were often cited by participants as their main reason for obtaining cultural artifacts via filesharing: they all expressed familiarity with the software needed to download files on filesharing networks, even when having to switch from a specific network to a different one (Kathy, for example, used to use Limewire and had recently switched to Bittorrent, when the former was shut down by a court order). Tawfiq noted:

> **[...] before, it was hard to download songs... people used to use Limewire... WHY would ANYONE use Limewire now?**

He went on to mention the perceived threat of pervasive presence of malware and viruses on Limewire as a significant security issue which could have discouraged users from filesharing, as opposed to the supposedly safer environment of Bittorrent, although he believed that it's up to users to be vigilant and to understand what they are doing, stating that only 'education' on some basic computing issues allows people to avoid security issues:

> **[...] they don't know the difference between a computer virus and a song... they don't know that music tracks need to have an MP3 extension, they click on EXE files and they get viruses. They're foolish.**

Not only was filesharing considered convenient as opposed to the industry-approved distribution channels on the web. Rosa's coursework involved the need to watch documentaries related to her final project's topic (vaudeville theatre and body performance in general): although the university's library had plenty of such video material available on DVD, she still found it easier to obtain these documentaries through filesharing.

Economic convenience was also mentioned: Kathy reported that she had around 30,000 songs in her iPod and around 7,000 in her laptop's iTunes and observed: '**if I had paid for each of them I'd probably not be at university**'. Whilst she also purchased music, Tawfiq relied exclusively on filesharing: he stated that he ever only bought a *cassette* long ago and had since only obtained music through copies and via filesharing; his single purchase was made for sentimental reasons: the album contained a song he loved and having earned some money to buy things for himself for the first time, he did actually purchase that cassette.

The shared view that filesharing is the normal way to obtain cultural artifacts was also reflected in how participant noted the illegality of what they were doing, only to dismiss it quickly: as shown above, Kathy couldn't afford to buy all the music she wanted and compensated any feelings of guilt with her occasional purchases, otherwise repeatedly chuckling to indicate that she was aware that her practices were not legal but that she didn't really see that as a problem: she considered them 'bad', but that arrangement was what worked for her. Rosa opened her account of her filesharing habits by acknowledging the illegality of her actions and simultaneously dismissing this ironically by stating emphatically: '**I... download [...]. I'm a pirate!! [laughs]**'.

Several important considerations at stake when discussing filesharing were essentially dismissed or outright ignored by the participants: the technical details of how to download, install and learn how to use the software needed for filesharing, the consciousness that, when using a p2p network such as Bittorrent, while downloading one is simultaneously also uploading content to other users and the resulting latent threat of 'being caught' sharing content — all these issues did not discourage these users. Drawing an analogy of terms and attitude with the 'Post-Open Source Software'[46] debate (Governor 2012), the participants' accounts relayed a shared feeling that, just half a decade after the initial fieldwork on filesharers' attitudes by Andersson Schwarz (2013), filesharing was seen as having become 'an emerging norm—if not even a new condition to media consumption' (Andersson Schwarz 2013, p1): a normal way to procure cultural and media artifacts for private consumption, even by users with no specific technical skills, effectively bypassing the many sides of the filesharing debates of the previous years as something that doesn't matter anymore, at the light of the actual widespread practice of 'post-p2p' filesharing.

Nevertheless, the infrastructural properties of p2p filesharing and the regulatory policies of the time were still shaping the context of these users' practices, despite having been hidden from their perception through black-boxing, and were therefore implicitly conditioning their computational agency. Firstly, the ability to simply 'download' content in which one is interested is made possible by the fact that other users are sharing this content at the same time as one's attempt to procure it through Bittorrent; that this has become increasingly easier is a combination of the network effects of a growing number of Internet users joining the Bittorrent network at any given time and of the increased availability of fast, always-on home Internet connections. Likewise, finding torrents and downloading them through a desktop application was indeed a rather simple task in early 2011 in the UK, but later court orders forcing major Internet Service Providers (ISPs) to block access to torrent index and tracker websites such as The Pirate Bay (which most UK ISPs started to block in the first half of 2012) made it necessary for users to know how to circumvent these blocks: however straightforward this may be in practice, it relegates non-technically minded users a few step away from the ability to 'simply download' which was reported, and expected, by my research participants.

While these issues highlight that infrastructural configurations beyond the sphere of awareness of users actively influence user agency, one must not rush to conclude, though, that limitations at the level of infrastructure can only be solved at the systemic level of the infrastructure itself: either in case of filtering/blocking of specific websites and types of content, or in case of scarce availability or outright non-availability of some niche content via p2p networks, some research participants discussed how they circumvented these

[46]The *Post-Open Source Software* label (Berkholz 2013; Fontana 2013; Governor 2012; Villa 2013) was coined in 2012 to describe the emerging pragmatic attitude of an increasing number of software developers who consider any publicly published source code, whether their own or work of others, as implicitly free to use. See detailed discussion of this in Chapter 7 as part of the analysis of hacker learning after the computational turn of the Internet.

infrastructural limitations by re-assembling specific aspects of their 'personal Internet'. Markus stated that attempts to block access to content on specific p2p networks would not have limited his filesharing ability, as he was already using VPN connections to disguise his real IP address and geographic location, as well as running a Bittorrent client on a virtual server hosted in a server farm rather than downloading content directly from his home broadband connection; he had been involved for some time in discussions with friends and online acquaintances about Internet censorship and had progressively built a rather sophisticated personal infrastructure with the help of technically knowledgeable friends. Kathy, instead, discussed how, learning from her father who loved jazz and found it difficult to gain access to some niche music albums via p2p networks, she would resort to searching for niche content on blogs where people share music albums as direct downloads.

Besides contributing to reshaping the ways in which users gain access to content, by making p2p filesharing an 'emerging norm', the computational turn of the Internet has also created possibilities for individual reconfigurations of the ways in which content is enjoyed. As with other domains of practices, the actual configurations observed varied in complexity, from simple purposeful sharing of content via SNSes (which, as discussed in Chapter 4, rely on considerable computation capacity) to coordination of content consumption across multiple digital devices. This space for individual choice in content consumption has been promptly acknowledged and exploited by content producers and device manufacturers, for example by designing *second screen* experiences, whereby TV watchers can access augmented content related to live broadcasts through the portable computational capacity of a tablet or smartphone device, often with the aim of increasing user engagement (and ultimately to create further opportunities for placement of advertisements). The fieldwork participants' accounts, however, focused mainly on individual-scale practices configured around the conditions under which content was enjoyed.

Christina described how she enjoyed watching movies 'together' with her boyfriend, even though he was living in her home city overseas while she was studying in London: they would agree over Facebook chat on a movie to watch, then count down and press the play button together, keeping a chat tab open in their web browsers to exchange thoughts on the movie as it was playing or just for banter. Whereas a programmed second screen experience would normally augment a movie watching experience with extra context, the configuration devised and cherished by Christina and her boyfriend worked by blending the online (the movie streamed over the Internet and their Facebook chat) with the offline — the intimacy of shutting off from their respective surroundings to enjoy 'being together' as they used to do while watching movies together in his home before she moved to London. Similarly, Tawfiq described the special value that a widespread practice — sharing YouTube videos and songs on friends' Facebook walls — had to him: as a student who tried to be 'always connected', carrying his laptop with him all day and keeping it on whenever he could get a wireless connection, he would not just 'share' a video he liked after randomly coming across it; instead, he would set aside some time every day searching for some special content that he anticipated a friend would appreciate, given what they were talking about at the time and how the friend was feeling like; he would then craft a brief sentence to dedicate the video or music track to the friend when posting it on their Facebook wall. To him, this was an important moment to show to a few important people that he was thinking about them every day even when they would not be able to meet in person, and that he cared about them.

6.3.2 Dealing with social norms online

As discussed in Chapter 2, early discourses around the Internet bore variously declined 'freedoms' as perceived distinctive traits of the 'online world'. This was a stage of the evolution of the Internet when most of the online interactions were happening through text (email, forums, instant messaging, website browsing) and textual environments (early MUDs and MOOs); moreover, home Internet connections were still relatively rare, so that early users wouldn't normally be able to extend in any systematic way through the Internet social connections they already had 'in real life', as most of their immediate acquaintances would not yet be 'connected to the Internet': accordingly, social interactions online were often essentially with strangers and were lacking the real life context that contemporary users of social network sites often enjoy as part of the experience of socialising through the Internet.

The lack of a broader social context to online interactions besides the textual elements which were sustaining most of these early online conversations, combined with the dominant technological determinism of the forming mainstream discourse around the Internet (Curran 2012), contributed to the development of much of the narratives of 'freedom' common amongst early Internet users. The commodification of broadband Internet access and the development of computational Read/Write Internet, however, contributed to shaping a very different context for the practices discussed by my research participants: on one hand, the pervasive use of social network sites evidenced the need to take into account normative issues connected with the reproduction online of existing social relationships, biases and inequalities, affecting to a higher degree more vulnerable users; on the other hand, the mounting awareness of privacy concerns and the even more pressing issues of government-sanctioned monitoring of citizens' activities online as revealed by the Wikileaks 'Cablegate' which had just begun at the time of my fieldwork, raised concerns that led some of my fieldwork participants to question the real extent of the freedoms that they discussed in their own accounts. In this section I analyze how research participants dealt with social norms and with constraints inscribed in technical forms, with the aim of developing an understanding of how computational agency was being reshaped at the time of my fieldwork, questioning the earlier ideals of online freedom and highlighting the traits of computationally-managed technologies which limited these students' capacity to reconfigure their lifeworld Internet; the next two chapters will then explore some hacker responses to the issues discussed here.

When discussing social norms connected to SNS practices, a recurring theme was the issue of whether and how to accept one's own parents and older relatives as 'friends' on Facebook: Christina, whose parents had recently divorced, felt that she needed to keep a boundary with her father as she wouldn't feel comfortable knowing that comments she shared with others could be read by him, whereas she accepted to connect with her mother — and was actually amused to observe her mother's avid use of Facebook, which she somehow related to her own, oriented at curating her own public image as well as specific connections that she was eager to further deepen offline. Rachel didn't accept either parents as friends on Facebook, in line with her long-standing attempt to regain, after having left the parental home, the independence that she had been negated as a female child in a rather socially conservative family: she maintained that she felt that her parents, still years after she started living by herself, would have criticised her life choices as these would emerge from her Facebook activity.

Another recurring theme was how to deal with 'unfriending' people on Facebook and how to keep some control over the content shared over time on Facebook: although unfriending people was seen as rude by most participants, those who had to deal with this issue took a rather pragmatic approach, which in their account helped them excuse their actions at least with themselves.

Peter lamented that, after signing up to Facebook and doing '**what everyone does... getting as many friends as possible**', he had gotten to a point where he was continuously seeing information about people that he didn't have any contacts with anymore: he therefore used '**a Firefox plugin that has a macro that you can program to delete all of your Facebook friends**' to schedule a mass unfriending of every contact and a parallel deletion of all the content he had shared on Facebook, then re-adding only the contacts meaningful to him, one by one: he shrank his network from more than 600 people to 101 friends at the time of our meeting. Similarly, Christina elaborated on a different but still pragmatic approach to unfriending: for her, besides day-to-day socialization, Facebook also had the important role of allowing her to quickly 'screen' students of her cohort when moving to a new university (first when she started her BA abroad, then when she moved to London for her MA) in order to find potential roommates with whom she may wanted to move in to halls of residence; accordingly, she tried to add as many of them as possible before starting her course through a snowballing strategy (first the ones who publicly made their status as students of the specific university publicly visible, then their friends): once she finished her undergraduate studies, most of the contacts accumulated for her screening process were effectively now useless, and she removed them without much afterthought from her Facebook account.

When analysing practices of early Internet users the possibility to disguise one's real identity and to experiment with alternative identity traits (e.g. gender, life circumstances, etc.) is often discussed. Building alternative identities is easier in environments — like the early text-only Internet forums and chat rooms — that lack significant cues and links to people's real identity in real life, as opposed to social network sites where the veridicality of an user's statements about themselves can be cross-checked by a multitude of other users who know them in real life; nevertheless, the practice of curating one's public profile with the aim of highlighting specific traits seen as desirable to have, rather than outright inventing whole new identities, is widespread across users of SNSes (Back, Stopfer, et al. 2010; boyd 2014; Tifferet and Vilnai-Yavetz 2014) and was mentioned by several of my fieldwork participants. For Christina, however, this practice went beyond the common aim of presenting an attractive *persona* and was instead more deeply connected to broader circumstances of her personal life: she discussed how she had recently felt she had lost her identity, while dealing with the trauma of her parents' divorce. After having embraced for a short period orthodox Judaism, which seemed to offer her a clear direction and references to rebuild her identity, she realised that the role she was expected to take within that religious-social context didn't match her life expectations, so she started experimenting more freely with different aspects of her public identity to try to find a new balance. Her 'obsessive' curation of her Facebook profile, in her account, was tightly coupled with this personal quest: she would post new status updates, notes and photos more than once every day, then often delete recent content that she felt didn't faithfully represent her perceived identity anymore. She was similarly obsessed with documenting most of her daily social life through photographs that she would then post on Facebook; later on she felt that this behaviour was excessive, realising how she was going on social outings in order to be able to take photos and share them: she soon '**weirded out**' and stopped taking her digital camera with her everywhere, preferring to deal with memories of events without the unrelenting documentary support of photographs.

SNSes were generally discussed in relation with the need to re-learn how to navigate social norms as sociality was now being enacted through a blended digital/non-digital environment: students' feelings about this ranged from frustration for the reproduction and amplification of pre-digital social norms to the excitement for the ability to experiment with less concerns about consequences, echoing some of the findings of boyd 2014. Student experiences, instead, were less mixed — and generally positive — when discussing practices of knowledge and cultural mediation, which were perceived as being reshaped through the Internet. Several students mentioned access to expert knowledge and information as relevant to their daily life: being able to quickly and independently find an answer to doubts they may have was seen as a precious attainment, especially when they would otherwise have had to resort to other people's expertise. This was particularly relevant for some female students dealing with digital technologies, which have been increasingly portrayed as a 'male domain' (Ensmenger 2010): Rachel, while setting up her home music studio, composed mainly of digital equipment, felt that, although she didn't have direct experience of these technologies, she could reliably seek information online or ask on forums, as opposed as having to rely on expert colleagues at her previous job; similarly, Kathy expressed frustration with the slowness of her recently-purchased second-hand MacBook laptop, and remarked that she would normally try to sort computer issues out by herself, looking for information online, before asking her boyfriend (who worked at an Apple store) for help.

Beyond accessibility of content, several students mentioned that they valued the ability to connect with publics without intermediaries over the Internet: Rachel welcomed this as a relief after having had to channel all her musical creativity through the strictures of the incumbent media industries, and stressed how she enjoyed being able to seek direct contact with her growing audience as a solo musician, as this was definitely hard but exhilaratingly empowering work. Similarly, Hye-jung had been developing her public profile as a travel writer by keeping a blog for over ten years, which allowed her to directly make new contacts with Internet users who started following her public writing. Peter, like Rachel, was experimenting with multiple Internet channels (a blog, Facebook, Twitter) through which he was publicising his bicycle travel diaries, building contacts and keeping in touch with people he had met through his long journey from the UK to India: his aim at the time of our meetings was to edit his travel diaries into a printed book, which he effectively published later on through an independent publishing platform, again avoiding what he felt the unnecessary mediation of middlemen such as publishing companies.

6.4 Reconfiguring computational infrastructure

In the previous two sections I focused on the strategies employed by students to meaningfully appropriate Internet affordances within everyday life, through personal reconfigurations of applications and practices as readily available, and incorporated into individual lifeworlds without any substantial reconfiguration of the code or the infrastructure behind them. In this last section, instead, I examine practices that involved a deeper relationship with software code and computational capacity, through the accounts of a few of my research participants who discussed different degrees of familiarity with technical layers of lifeworld Internet not immediately accessible to users without some technical knowledge. Understanding, manipulating and reconfiguring parts of the technical substratum to their practices by operating directly on code and infrastructure was an important part of these students' experience, highlighting a deeper engagement with computation and with the political implications of the act of seeking control over parts of lifeworld Internet that are often adopted by common users in their black-boxed form, as readily available services and affordances. In order to understand the practices and motivations of these students within the dissertation's focus on the nexus of computational agency and lifeworld Internet, I will first take a small step back to analyze the stories of two students whose practices appeared situated at opposite ends of an ideal spectrum of computational agency, with the aim of complicating the assumption that a higher degree of computational agency is desirable a priori, independently of other factors that contribute to make an individual's lifeworld Internet practices meaningful and able to satisfy their own immediate needs.

6.4.1 Read/Write practices and meaning: a complicated relationship

Two of my first few fieldwork interviews started as frustrating and somewhat awkward conversations with two students who professed very little interest in any of the Read/Write or even social network practices, making me wonder how little useful information I could gather from these interviews. When reviewing the whole set of interviews, however, one of the most visible patterns was a non-linear relationship between sophistication of Read/Write practices and reported satisfaction for the support that students found through their own use of the Internet in everyday life. Although students who reconfigured their Internet environment in more complex ways were generally also linking their digital practices to social interactions and political activities that appeared more intense than those of students who expressed satisfaction for a more basic Internet use, students on both ends of the spectrum of computational agency felt that their Internet activity was overall responding to their needs: I didn't encounter substantial regrets for the inability to achieve desired objectives by exploiting Internet affordances — if at all, some concerns were voiced by students who were more active online and also more aware of the limitations involved in their practices: whether because of their own lack of knowledge on how to reconfigure their Internet environment in even more advanced ways or because of external (technical, economic, policy) constraints.

On one hand, it could be expected that users who are less aware of sophisticated uses of Internet affordances would not feel that they are 'missing out' whilst not including these practices in their own everyday life; their overall satisfaction with practices that involve less control over their Internet environment, however, can also be better understood by challenging the primacy of *content* in Web 2.0 narratives. Whereas the Read/Write trait of Web 2.0 involves production, sharing of (and collaboration over) digital artifacts that can generally be considered 'content', my fieldwork participants highlighted that what actually mattered within their everyday life was production of *meaning*, which often involves various acts of manipulation of content (creation, sharing, etc.) but is both analytically distinct from these and more immediately relevant to each user's lifeworld. Fragments of the stories of two research participants exemplified this distinction: Ian and Markus described practices differing widely in terms of technical expertise, political engagement and 'read vs write' configurations; however, both highlighted the value they placed in their close connection (personal for Ian, political for Markus) with the local community. Using the Internet as part of their strategy to develop their local connection, both meaningfully exploited the digital as a way to augment their offline sociality.

Ian repeatedly and almost apologetically stressed how he saw himself as a 'basic' Internet user, mainly 'going online' to browse news on mainstream media and to read local news and comments on blogs focused on the area around Goldsmiths; Markus had been running workshops in the local community to teach people how to plan, shoot, edit and share short documentary videos as part of his practice-based PhD course, with the aim to help give voice to underrepresented concerns and to promote accounts critical of the mainstream discourse of 'regeneration' of the poorest areas of the neighbourhood. Ian's interest in the local community was rooted in his local upbringing: for him, following local blogs was a way to keep up to date with local news and conversations with minimal effort, but was also a way to discover new cafés (the area has seen a lively development of such venues, complementing the array of existing street food outlets) where he could gather with his local friends. Markus, instead, as an overseas student with a history of engagement in community media activism (street-TV projects in Italy, community TV stations in Jamaica) started interfacing with the local opposition to the agendas of real estate developers masked through discourses of 'regeneration' as part of his documentary filmmaking research project, which revolved around an Internet-based, participatory television channel of the local community. Whereas Markus contributed many hours of his time every week to the curation of user-contributed content, to run workshops and to promote his web-based project, Ian's engagement was undeniably of a different kind, and he explicitly stated that he was interested in learning about local developments but not in taking part in activist projects around local issues, besides occasionally signing petitions online. In both cases, however, a sense of somehow belonging to the local community motivated their interaction — intensely Read/Write for Markus, mostly read-only for Ian — with content produced locally and published over the Internet.

It would be tempting to dismiss Ian's use of the Internet as that of a self-centered, politically unengaged young person, blissfully unaware of (or unconcerned by) the violent processes of displacement of the poorest dwellers of the local community and of destruction of family businesses through the ongoing gentrification; whereas this could be a valid remark, judging my participants' political engagement is not the point here: these two stories (chosen because of the common thread of connection with the local community), like many others related by the students I interviewed, highlight two key traits recurring throughout my fieldwork: firstly, the fact that students were primarily concerned with a lifeworld domain that included basic, everyday concerns, which were addressed through a combination of digital and non-digital practices. For some students this domain included mainly their private sphere and basic needs and wants, whereas for others their community, social or political engagement was an important part of their everyday concerns and, relatedly, of their Internet practices. The domain of activity referred to as 'lifeworld Internet' through this dissertation can be better understood by considering the everyday unfolding of mostly mundane and unglamorous 'stuff' in daily life and the individual strategies employed to re-configure available Internet technology in support of it. This may well include using the Internet for apparently idle and selfish aims, but the relevance of the fabric of the every-day — whatever its social worth — to each person's lifeworld is what makes the analysis of computational agency within the domain of lifeworld Internet pressing and urgent, along-side the analyses of political engagement through the Internet widely developed in existing literature, whose societal value can be more immediately acknowledged.

Secondly, and relatedly, the emphasis on production of content stemming from early Web 2.0 narratives needs to be complemented with the exploration of what matters to Internet users in everyday life, including the untangible but nevertheless very material concerns revolving around meaning: whereas the project analyzed in the previous chapter—following Web 2.0 assumptions—failed to enlist users *also* because of a simplistic reduction of production on the Internet to production of artifactual content—whether text, photos, video or some data–based representation of a social graph—any intervention (such as those discussed in the two following chapters) aimed at subtracting lifeworld Internet from the ongoing centralization under the control of a few large corporations needs to employ a more complex understanding of production focused on the richness of users' lifeworlds, to which ethnographic research can contribute substantially.

6.4.2 Hacking lifeworld Internet: advanced practices

As discussed in the first section of this chapter, none of the students interviewed made any substantial use of the computational interoperability features of web applications to create Web 2.0-style mashups; several users, however, described how they were nevertheless re-configuring their lifeworld Internet through the creation of computational functionality aimed at responding to their specific needs: this was done by 'hacking' to various degrees the affordances available to them beyond the *designed* flexibility and ability to integrate them. These interventions ranged from small-scale technical reconfigurations (or 'hacks') to complex practices involving setting up personal servers and virtual machines and managing through them services (for example, their personal email workflow) for which other students would rely on services managed by Google, Apple, Facebook or other companies. Analytically, these interventions could be placed on a continuum of computational agency that extends from basic, occasional 'hacking' to the complex hacker interventions discussed in the following two chapters. In every case, however, the object of these reconfigurations was the *technical form* itself, rather than textual or media *content* manipulated through technology, which is the reason for discussing them together here.

Emir, one of the only two students who claimed to be making a read-only use of the Internet, linked the low complexity of his Internet practices to a severe lack of time for anything else than working on the final revisions of his doctoral dissertation. He had in fact been involved in Internet radio broadcasting before starting his PhD course, which gave him first-hand experience of content production and distribution on the Internet, but he stressed that what he saw as the most vital Read/Write function of the Internet at the time of our fieldwork meetings was simply to allow him to distribute across the Internet several copies of any new versions of his dissertation draft in the event of an unexpected concurrent failure of his laptop and of his data backups on the university's network: he mentioned that his fear that a solar magnetic storm may damage computing equipment across areas of the planet had led him to realize that he could simply send successive drafts of his dissertation to himself as email attachments as a form of automatic data replication, assuming that different copies of his email messages would be stored across geographically distributed data centres operated by Google for their Gmail service, therefore providing a higher level of redundancy than the locally stored backups offered by the university's IT services department. Whereas Emir would certainly not have been the first to exploit such strategy, he expressed pride for what he considered a simple but ingenious hack, and which, most importantly, responded to his single most pressing everyday concern at the time.

A small subset of research participants — those with some skills in software development or systems administration, or at least an interest in these topics — on the other hand described more complex practices, explicitly — rather than incidentally — involving a deeper understanding of the infrastructure behind the visible affordances that they were ultimately interested in using. Through their practices, they reshaped their lifeworld Internet environment in highly personal ways, modelling their interventions after specific life, study or professional circumstances. All these students reported that their efforts were made possible by the use of free/open source software which, specifically, allowed them to:

1. learn about possible ways of reconfiguring their Internet environment to respond to specific needs;
2. learn how to do this and
3. effectively reconfigure their Internet environment.

'Hackability' of software affordances and the ability to inspect what software does and — when it fails — why it fails, were the practical benefits that these students saw in the use of free/open source software. Moreover, each of them was actively embracing the political values behind free software as inscribed in the four essential freedoms of the GNU GPL license: the ability to run software for any purpose and to modify it, and to help others by letting them use one's contributions to free software projects.

Markus' PhD project revolved around the use of free software and the creation of audio/video content to be released under free culture licenses: as one of his project's main aim was to empower ordinary users from a deprived neighbourhood to publish video content on the Internet, he realised that he could only rely on free software (specifically, a customised Linux distribution targeted at audio/video production and able to run on old, underpowered computers) if he wanted to avoid complex and costly software licensing issues; moreover, he was actively collaborating with several free software projects in order to develop a distributed yet easy to use Internet infrastructure that could be used by his project's participants to collaboratively edit, host and publish video on the Internet without having to rely on proprietary cloud services. His disdain for proprietary, closed infrastructures was similarly evident in his personal Internet setup: he had learnt how to manage his own web server, on which he also run his personal mail server, in order to safeguard the privacy of his own Internet activities; likewise, as he regularly worked from different locations and didn't want to resort to proprietary cloud services to be able to have his data and applications available through the Internet from anywhere, he started setting up, with the help of friends, a simple personal cloud infrastructure, which in turn he then shared with the friends who helped him setting it up and with other collaborators similarly concerned about the privacy and safety of their data.

Like Markus, Inaki didn't have a background in computing (he was a Sociology student) but had developed through time a very personal Internet infrastructure to be able to autonomously configure some services that he saw as an important part of his everyday life and didn't want to entrust to proprietary providers. Before moving to London for his Masters course, he had been working part-time as a builder and as a postman in his town in the Basque country in order to save for his postgraduate studies: while delivering mail across town, he realised that several elderly and vulnerable people lived alone and didn't have much support around them, so he soon started visiting each of them every day even when there was no mail to be delivered, just for a chat, to check in on them and make sure they were fine and able to contact relatives, friends or neighbours for any support they may have needed, such as having someone deliver groceries to their flat if they were unable to go out for shopping. As these people started relying on his daily visit and told him he would be missed when he was about to leave for London, he set up a VoIP account with a local provider, associating a local landline number to it, and through his own Internet server he was able to redirect calls received on his Basque country landline number to wherever he would be, as long as he had his laptop or smartphone with him. He didn't want to rely on Skype as, he stated '**[...] nobody knows who exactly can access the content of Skype calls, maybe they're monitored [...] That would mean somehow, you know, betraying my friends who may just want to call me for a chat [...] They don't even know what the Internet is, they just dial a number on their phone and expect to hear my voice, I can't just let their calls go through a black hole, even if nobody will care about what we say.**' As he was managing the full configuration of his own VoIP infrastructure on his own Internet server, he soon ended up setting up VoIP accounts for his parents on their own computer at their home, through which they could speak at no cost and without relying on proprietary services. Soon, he said, '**[...] the Internet bug started biting me [...] I enjoyed the freedom and flexibility of having my own little infrastructure, it's private and all that, and bit by bit I moved there my email inbox, my storage, my remote backups... It feels good, if it breaks you can fix it, and you know you are in charge.**'

Peter described how he had been setting up self-hosted blogs for his own public travel diaries and for other projects, capitalising on earlier work he had done as a self-taught web developer after finishing his undergraduate studies. He discussed how important the Internet had been to him as a platform where he felt that he could freely publish his travel diaries and, more importantly, his reflections and meditations about living a nomadic life depleted of much material possessions and about the casual meetings he had with strangers through his cycling journey, hoping that the values that informed his long bicycle journey would inspire others to question their own values while leading a life in a stable place with solid routines throughout their days. Accordingly, he was very concerned about what he saw as threats to the future sustainability of an independent, open Internet, and discussed how he believed that free/open source software was essential to an open Internet. Besides setting up his own blogs, he hadn't yet taken on more complex, infrastructural projects such as those discussed by Markus and Inaki, but through the practical projects of his Design and Critical Practice MA he was learning how to develop web apps that could engage users in questioning their ordinary spaces in innovative ways.

Markus discussed in detail several concerns he had about ordinary Internet users becoming increasingly reliant on proprietary services, that he saw as driven solely by profit and ultimately unaccountable while dealing with often very personal and intimate data; similar concerns, although in less detail, were expressed by Inaki and Peter. These students could all be considered 'ordinary users', who had taught themselves how to reassemble parts of their lifeworld Internet beyond what they saw as essentially false choice given by proprietary vendors through the flexibility of services designed to capture user attention; they successfully developed, with the help of information available online and with occasional support from friends, alternatives to mainstream Internet infrastructure, at a micro scale but in ways highly relevant to their everyday concerns. Their interventions were somehow limited in their impact beyond their own use and that of a few close friends or relatives, as opposed — for example — to the work of developers working on free software web apps for decentralised social networks such as the diaspora* project[47] (to whose crowdfunding efforts Markus had contributed), yet showed how concerns about an open Internet could become a central part of some users' everyday experience of the Internet. Finally, both Markus, Peter and Inaki openly challenged the narratives of freedom that they considered still present in mainstream discourse about the Internet, acknowledging that user agency is always limited by many factors beyond the single user's reach, which could in part be circumvented through alternatives to mainstream social network sites and to mainstream Internet infrastructure more generally, but that would still shape the overall space of possibilities and of limits that apply to most ordinary Internet users.

[47] https://diasporafoundation.org/.

Chapter 7

Hackers as recursive public: constitution of computational mediators

> ”The best minds of my generation are thinking about how to make people click ads”. (Jeff Hammerbacher, quoted in Vance 2011)

7.1 Introduction

This chapter analyzes the constitution of hackers as an increasingly visible ’recursive public’ (Kelty 2008), acting as computational mediators at the interface between the technical and the social, establishing and expanding spaces where user and developer discourses are able to effectively shape the development of technical form, and developing actual affordances aimed at addressing the power imbalances inscribed in mainstream Internet such as those analyzed in the previous two empirical chapters.

In chapters 5 and 6 my focus has been the analysis of reconfigurations of computational agency within the domain of lifeworld Internet, at the local scale of my fieldwork sites. By deconstructing these reconfigurations, as experienced through my Web 2.0 development project and by the students I interviewed for the second phase of my fieldwork, I have attempted to show that users' computational agency — what they could make of lifeworld Internet, and to what extent and how they could reconfigure lifeworld Internet affordances, vis–à–vis the early promises of choice and empowerment carried by Web 2.0 narratives — has been deeply affected by systems of power relations progressively inscribed in the technical form of mainstream Internet. In order to ground this analysis in the materiality of Internet affordances and infrastructure, in Chapter 4 I articulated the analytical framework of computational turn of the Internet and I sought to identify the key traits of computational complexity leveraged by large corporations to build and control mainstream Internet infrastructure at a global scale, while suggesting that these same traits are potentially helpful and meaningful for the efforts of independent developers who wish to engage in counter-narratives of lifeworld Internet and to develop alternative rationalizations able to sustain different and more open forms of computational agency for end users.

Whereas computational capacity may present determinate degrees of openness to exploitation for progressive ends, however, it is obviously only through intentional human labour that affordances are appropriated and reshaped and that specific configurations are effectively attained. Accordingly, in these two final empirical chapters I shift my focus to actual counter-narratives and alternative rationalizations, relating back to the affordances and practices relevant to the user experiences observed throughout my local fieldwork, in order to examine how these alternative rationalizations may enable different configurations of computational agency that overcome some of the external constraints analyzed in the two previous chapters, allowing common users to exert control over their lifeworld Internet beyond the range of possibilities available through Internet affordances and related narratives as determined by a largely capital-driven mainstream development.

Specifically, this chapter looks at the junctures within the complex social and technical structures involved in the production of Read/Write Internet where alternatives to the hegemonic implementations are discussed, made public and developed: these spaces and conditions in which counter-narratives can evolve and inform the actual reshaping of Internet affordances are here seen as an often invisible middle ground between the visible domains of corporate Internet services and of end users. On a local scale, in the context of my fieldwork I observed cases in which the boundary between software producers and consumers was blurred to some extent (the accounts of Peter, Markus and Inaki analysed in Chapter 6 provide the most compelling insights of this), thanks to the malleability of Internet infrastructure and to the ability to gain expertise relatively easily (for example through access to online documentation and discussion forums or through support from friends or acquaintances) in order to be able to exploit this malleability. At a global scale, public discourses of hackers involved in counter-narratives highlight the importance of recent (late 2000s-onwards) changes in negotiations of technical form and in reproduction of knowledge as the main factors that are supporting the development of a new hacker 'recursive public' (Kelty 2008) as a class of computational mediators able to effect change through their labour. Understanding the constitution of this recursive public is instrumental to the analysis (developed in the following chapter) of how hackers effectively articulate actual affordances developed as alternative rationalizations to the hegemonic technical form of mainstream Internet. The constitution of a new global hacker class as recursive public is here seen as necessary condition for the actual attainment of the possible progressive ends of increased computational capacity available through the computational turn of the Internet.

The empirical material on which this chapter and the following one are based is constituted by public hacker discourses (recordings of presentations at technical conferences, discussion forums, technical blogs, publicly available source code): I have been fully immersed in these discourses for almost two decades as part of my daily professional learning and I have revisited recent (post-2010) materials as part of my ethnographic exploration aimed at understanding computational agency within the scope of my dissertation. Whereas the space constraints of the present work allow me to use directly only a limited amount of literal quotes from these discourses, my focus on computational mediation and reproduction of knowledge throughout the present chapter reflects what I observed as a growing consensus and relevance within the vast corpus of hacker discourses on which I relied for my research. A detailed overview of my source materials is provided in Appendix B.

This chapter is composed of three sections. In the *first section* I look at strategies through which hackers recursively inscribe in technical infrastructure the ability to take into account context-specific user needs rather than forcing these needs to be reframed within the limited options available under mainstream technical configurations, as was largely the case until the late 2000s. Whereas Internet software engineering practices were initially polarized within a limited range of widely adopted configurations, the computational turn of the Internet coincided with a shift towards increasing diversity in how Internet software is built and in how best practices and standards are progressively defined, through feedback from and debates between developers. The aim is to identify how complexity of 'real world' use cases is reflected in computational complexity through the public mediation of hackers, as opposed to earlier top-down approaches (as exemplified by the case study of Chapter 5). In the *second section* I focus on hacker learning and on reproduction of hacker knowledge: my argument is that through the wide accessibility of learning resources (documentation, tutorials, recordings of presentations at technical conferences, discussion forums, publicly available source code), hacker knowledge is being accumulated, reviewed and reproduced at a global scale, resulting in improved opportunities for hackers to learn from quality resources and to attain capacity to develop meaningful software. In the *third section* I look at how hackers are progressively constituting a new global recursive public intent in understanding the implications of its own capacity to operate at the intersection of the technical and the social through the role of computation.

7.2 Inscribing recursion in computational infrastructure

In this section I look at how hackers have successfully appropriated processes through which the relevance of different publics is incorporated into the development of technical form, through best practices in software engineering and formal recommendations, standards and protocols. This appropriation is the first strategy analyzed here through which hackers have been constituting themselves as a recursive public 'vitally concerned with the material and practical maintenance and modification of the technical, legal, practical, and conceptual means of its own existence as a public' (Kelty 2008, p3): by recursively developing technical form, by defining best practices of Internet software engineering, by ensuring that hacker discourses are taken into account throughout the development of standards and protocols that define the formal foundations of Internet affordances, hackers act as computational mediators that open up spaces for the needs of their software's users and for their own needs (as developers and, often, recursively as users of their own or of other hackers' software) to be incorporated into technical form.

7.2.1 Negotiating computational complexity from context diversity

Alongside the computational turn of the Internet discussed in Chapter 4, the years 2010-2011 marked a noticeable shift in practices and theories of internet software engineering, thanks to the increasing availability of software libraries, tools and methods that rapidly innovated web development compared to the relatively slow evolution of tools and methods over the previous decade.

Throughout the core Web 2.0 years (early 2000s to 2010), for example, a single dominant software architecture was used for the development of most web applications, based on the Model-View-Controller (MVC) pattern; this pattern was devised in 1976 and formally described in 1988 (Krasner and Pope 1988) and in those pre-Internet years it was originally only employed to organize interactions and data flows within seminal graphical user interfaces, whereas its dominant role in web development can be ascribed to the success gained by early web application frameworks[48] based on this pattern, such as Ruby on Rails, Spring and Catalyst, which were used as a base software layer by most Web 2.0 applications. The 3D Graduate application discussed in Chapter 5, notably, relied on the MVC pattern (and on the Catalyst framework as practical implementation of it) like most Web 2.0 applications of the time.

As a computational layout defining the interface between data (the 'model'), computation (the 'controller') and user interface (the 'view'), the MVC pattern inscribed in Web 2.0

[48] A *web application framework* is a combination of software libraries and associated software engineering methods which provide essential features upon which web applications depend: as most of the low-level interactions between a user's browser and a web application (such as receiving and processing user requests, fetching data from backend databases in response to these requests and creating the web pages served back to users) can be abstracted from the application-specific context, developers normally rely on a web application framework to take care of these interactions, in order to be able to focus on the core features such as app-specific content and workflows.

applications a structure that, combined with technical limitations of web infrastructure available to software engineers in the years prior to 2010[49], relegated to each web application's provider an authoritative role as gatekeeper of content (whether built by the provider or generated by users) and of the range of allowed possibilities to describe the meaning of content, limiting at the level of infrastructure the space left for end users to inscribe their personal interpretation of relationships between items of content. More 'daring' designs, trying to embrace semantic web technologies of the time to return some of this freedom to users, typically faced both technical limitations (as most of the readily available software libraries inscribed inflexibly the MVC pattern that successfully sustained top-down knowledge representations in web apps) and the inability to effectively act as translators between heterogeneous groups of stakeholders. As the lead developer of such an early web app candidly described in a private email to me, echoing common frustrations I encountered during my own development work analyzed in Chapter 5:

> The story behind what happened with [software project], [event series using this software] and [website running this software] is a bit of a tragic one. [...] What we were building was [...] dependent on being able to pull in and connect information from various other websites [...]. We took far too much for granted about our ability to convince a network of professional curators, used to having ultimate control over how information is presented and arranged, about an open, aggregator based approach. So we presented them with a rough prototype, and their reaction was, 'there are seven different descriptions of this event! which one is canonical? this is unacceptable!', and 'how can we expect people to visit several other sites and "input data", not knowing how it will wind up in the final "product"?' (Walsh 2006)

Another technical configuration consistently dominant (so much so that this configuration rapidly started being referred to through its acronym) throughout the core Web 2.0 years was the LAMP[50] 'stack' of software employed for the execution and delivery to users of Web 2.0 applications. Although several alternatives to each of the components of the LAMP stack existed, this specific technical configuration remained dominant thanks to the broad availability of documentation, tutorials, online resources and professional

[49] As an example, early Web 2.0 projects would have very limited options if they needed to reliably integrate data across very large Internet infrastructures possibly spanning over organizational boundaries: the 'model' of most MVC implementations was in fact often limited for technical reasons to a pool of database servers, with little ability to keep connections with external data constantly updated, unlike later distributed 'NoSQL' databases which have increasingly been updated to accommodate wider ranges of use cases rather than forcing development projects to model their data needs around the tools available.

expertise focused on it. Through its dominance, it inscribed in much of Web 2.0 development top-down flows of knowledge in a similar way to the MVC pattern discussed above, by providing a reliable platform that relegated computational complexity to servers, while clients (web browsers) would mainly consume content sent to them and contribute user data stored back on servers.

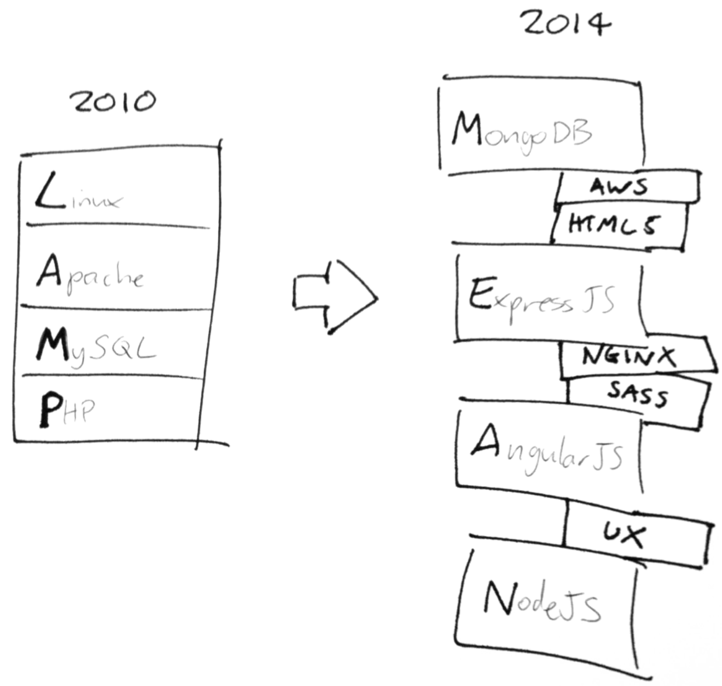


Figure 7.1: Schematic chart of software infrastructure for a typical mid-2000s Web 2.0 app versus a typical post-computational turn web app
Schematic sketch illustrating the transition from the traditional 'LAMP' (Linux, Apache, MySQL, PHP-Perl-Python) core set of technologies upon which most pre-2010 Internet projects relied, to the complex network of interactions between more, and more diverse, technological and organizational domains typical of post-2010 Internet projects (illustration from "The Myth of the Full-stack Developer", Shora 2014).

In contrast to the relatively slow pace of innovation in these early software engineering practices that reproduced pre-Internet patterns of authority over content, recent hacker discourses (primarily engineering blogs of startups and technical blogs of individual hackers) highlight an increasing diversity in approaches, often emerging at a fast pace as ex-

[50]Acronym for Linux/Apache/MySQL/PHP (or Perl or Python), after the typical 'stack' of layers of specific technologies used, including operating system (Linux), web server software (Apache), database (MySQL) and the programming languages used to develop MVC web applications (one of PHP, Perl or Python).

periments to adapt existing tools and practices to specific contexts or to create new ones when none of the available ones appear to be suitable foundations on which to build software intimately shaped around the peculiarities of a given real-world use case. Threads such as the 'Show HN'[51] ones on the discussion forum Hacker News, through which developers showcase their own (often still experimental) software, often highlight ingenious ways to reconfigure existing software, libraries, protocols or best practices: the discussions unfolding through these threads (and similar ones on development subforums on Reddit or other hacker discussion venues) constitute often enlightening informal peer reviews of others' work, with topics ranging from technical merit, discussions of improvements over existing implementations, critique of the aims of the software and offers for contributions (where the discussed code is released as FLOSS).

As the figure above shows, through the interpretation of developer Andy Shora (Shora 2014), the substantial uniformity of the popular LAMP stack introduced above has been questioned through the introduction of many competing 'stacks', some of which have been adopted as best practices in specific contexts while a majority constitute niche experiments or very specific interpretations of how to assemble technical form to respond to peculiar needs. Tutorials and informal accounts on developer blogs show how different configurations are intentionally mapped to the scenarios in which developers operate, highlighting a common trend of shaping technical infrastructure using smaller components available as FLOSS libraries and tools, rather than adopting what developers often call 'monolithic' stacks (such as the LAMP one), trying to reframe individual context-specific requirements within the features and limits of these stacks. What can be observed is a transition from generic tutorials explaining how to develop software within previously widely accepted best practices to discussions that highlight how specific projects have found meaningful ways to assemble technical form: in turn, these discussions contribute to the shaping of the hacker recursive public, making bottom-up processes of affordance–building a new normal way of responding to specifications emerging from user research. Whereas it is important to keep an analytical distinction between 'users' and 'end users' here (the reference 'users' of a social network site developed in a corporate context would often primar-

[51] 'Show Hacker News'.

ily be the company's stakeholders rather than the SNS' actual end users), the point of this section is to analyse the progressive questioning of general best practices and the development of approaches sensitive to actual use contexts. Although these approaches could be used both for centralization ends (in the case of typical mainstream Internet services), they also constitute the foundations for workflows of development of alternative rationalizations, which (as will be discussed in the next chapter) often start from the questioning of hegemonic rationalizations and therefore need to ensure that technical form effectively responds to the needs of the reference users, limiting the reproduction of assumptions and power relations inscribed in top-down best practices and the corresponding translations into software.

7.2.2 Material grounding of new Internet standards

Besides reconfigurations of technical form as just discussed, developer discourses highlight the relevance of changes throughout the governance layer of Internet standards to the capacity to develop alternative rationalizations. Fundamental web standards such as those collectively grouped under the HTML5 and CSS3 labels have been developed since the mid-2000s through negotiations, discussions and processes vastly different from those of previous versions of these specifications and have taken into account web developers' discourses directly, thereby making Read/Write Internet more directly *writable* at its lowest levels (those of infrastructural protocols) than it had been in the past; whereas earlier versions of web standards, such as the HTML markup language specifications up to its version 4[52], had been developed through a consensus process within the World Wide Web Consortium (W3C), typically trying to balance requests of corporate members such as major web companies, development of the HTML5 specification[53] was instead mainly carried out by an informal group of individual developers (Web Hypertext Application Technology Working Group — WHATWG), initially from major browser vendors, concerned about what they perceived as the excessive formality and overly long timeframes of the W3C's processes (WHATWG 2015), which made it impossible for browsers to provide functionality needed for modern web applications and user experiences through shared standards. Rather than trying to orchestrate a complex equilibrium of variously misaligned corporate interests within the formal consensus-driven processes of W3C, the WHATWG appointed a single *informed editor*, Ian Hickson, as the driving force behind the evolving HTML5 specification.

The key difference, in the context of reassembling Read/Write Internet, was that requests and feedback of a much larger group of developers were taken into account by Hickson through his successive drafts of the various parts of the in-progress specification: rather than gathering technical and legal representatives from corporations and asking them to put forward their various proposals, Hickson drafted his own proposals interpreting through his expertise and professional judgement what he considered to be, at various points in time, best practices and technical needs amongst the web developers community at large, stemming from the material details of work being developed rather than from purely theoretical debates about expected benefits of specific designs: discussions on technical merits of competing designs and implementations, often between individual developers debating in forums or through discussions carried out across personal blogs, informed the evolving specification as much as the mix of technical, commercial and political interests from browser vendors and large media companies had been doing in previous formal consensus-driven processes.

For the first time since the initial design of web protocols by Tim Berners-Lee in the early 1990s, developers had a substantial opportunity to contribute to reassembling the infrastructure of the Web: although this shift in power relations wasn't without struggles, at times acrimonious debates and governance issues (Jaffe 2014), it resulted in much broader opportunities to experiment for web developers, makers and Internet of Things tinkerers, which in turn made possible the development of the alternative configurations analyzed in the following sections of this chapter.

The overall change of approach in negotiations related to the development of Internet infrastructure is inscribed in the W3C's own 'HTML Design Principles':

> **3.2. Priority of Constituencies** In case of conflict, consider users over

[52] HTML 4.01 became a final recommendation in December 1999 (W3C 1999).

[53] Work on a new version of the HTML language started in 2004 but only gained momentum and mainstream visibility around 2009, when the Firefox browser (version 3.5) and the Google Chromium project, on which the Google Chrome browser is based, started implementing some features of the draft HTML5 language, allowing web developers to actively experiment with features under development.

> authors over implementors over specifiers over theoretical purity. In other words costs or difficulties to the user should be given more weight than costs to authors; which in turn should be given more weight than costs to implementors; which should be given more weight than costs to authors of the spec itself, which should be given more weight than those proposing changes for theoretical reasons alone. (W3C 2007)

Even within the more open approach taken by the WHATWG, the ability for minority constituencies (relevant social groups), such as those of developers working on niche projects, to influence development of standards specifications should however not be overstated: overall steering of specifications is often still carried out by experts working for large corporations (Ian Hickson is a Google employee), and the W3C still plays a role in endorsing specific versions of the evolving HTML specification through a consensus process, albeit this has been reshaped to be more responsive to the need, highlighted by the WHATWG experience, to model protocol development discussions and decisions around the requests emerging from of a broader pool of constituencies, some of which are in much closer contact with the material practices, motivations and struggles of end users than product managers and developers from a narrow group of browser vendors (Jaffe 2014).

7.3 Read/Write hacker learning and reproduction of hacker knowledge

Equilibria between relevant social groups have been transformed throughout the computational turn of the Internet alongside dimensions other than the governance ones just discussed. Evolution of the materiality of hacker learning and of reproduction of hacker knowledge constitutes one of such essential traits as emerging from developers' own self-reflective discourses: how gaining a progressively growing understanding of Read/Write Internet infrastructure has evolved in parallel with recent developments of technologies, and how this reshaped learning landscape has affected the ability for non-technically trained users to effectively operate and hack on Read/Write infrastructure.

Technical evolution[54] and more representative governance, indeed, would not by themselves necessarily lead to the opening of spaces for development of alternatives, unless ways to exploit these changes for personal or social benefit become accessible to concerned actors (Hippel 2005). Relevant developments in the ways in which hacker learning is configured, however, happened concurrently to technical and governance changes, helping both to attract proportionally increasing numbers of individuals to web and Internet development, as well as enabling a wider and faster circulation of discussions and practices related to alternatives to mainstream Internet configurations. Nevertheless, struggles about lack of diversity within Internet developers communities have not resulted yet in a significant redressing of long-standing inequalities (especially in terms of gender, class and ethnicity), although they have partly succeeded in making these diversity issues visible amongst developers.

Traditionally, hacker learning has been a process situated in three main environments: formal learning settings (such as university degrees or professional programmes); hacker meetings/conferences; and the private home or work place of individuals (Coleman 2012a; Kelty 2008; Levy 2001). Qualitative surveys often highlight how informal learning is reported as being prevalent by hackers (e.g. Case 2015) while recent government reviews in countries such as the UK (UK Government Migration Advisory Committee 2015) and the US (in the government and private sectors: Xue and Larson 2015) identified severe shortages of skilled IT workers, which are both attributed to outdated curricula and linked to a wider lack of appreciation for the importance of *computational thinking* as a critical attitude and a broad set of skills required to address practical issues through software. My own reviews of professional magazines targeted at web developers, as well as my experience teaching web development classes in an academic setting, highlighted similar concerns: a general recognition of the value of foundational courses is almost invariably complemented by the notion that practical courses struggle to stay relevant when tools, methods and strategies used in real-world contexts often become outdated within a very short timespan:

> The problem is that it takes such a long time to create a curriculum that the technology has moved on by the time it's delivered. My gut says it's the

[54]Evolution is here not meant in a linearly deterministic and optimistic sense, but rather as development of an increasingly *broader and more diverse* array of *competing strategies* to address user needs efficiently with the aid of software, such as improved understanding of ways to optimize different algorithms and processes implemented through software, more efficient development workflows enabling faster feedback cycles and therefore increased ability for developers to experiment with alternative strategies to meet functional software design goals, as well as faster and more power-efficient computational capabilities for Internet-connected devices.

> fringe initiatives, the code clubs and coding dojos run by people like me that are going to make the difference. (Remy Sharp, in Combrinck 2012)

In the same discussions, more experienced professionals providing guidance to beginning developers stress how much they value field experience when making hiring decisions, up to suggesting that developers' public code contributions hosted on GitHub are one of the most important talent discovery and assessment vehicles (Weiss 2012, although other commentators such as Coglan 2013 highlight how relying on developers' public contributions risks reproducing current inequalities in job opportunities for underrepresented groups in IT). On the other hand, informal learning, whether self-directed or peer-supported, has traditionally been not only one of the main ways for beginning developers to gain experience and skills, but more importantly an opportunity for them to become part of groups of peers in what are often very geographically sparse communities.

Through the following sub-sections, I focus on distinct contexts through which hacker learning and knowledge reproduction are reconfigured in ways that hackers themselves consider meaningful for the progressive development of computational agency; the reliance of these contexts on computation (through algorithmic management of discussion and of relevance of content, through visibility and ability to manipulate source code) highlights how computational agency is here specifically analyzed as both the ability for hackers to appropriate computational affordances for personal or group aims (in this case, improving one own's understanding of technologies and of the role of computation as layer of mediation between the technical and the social) and to transform computational affordances through the knowledge gained. At the same time, the focus on specific sites (Stack Overflow, the Stack Exchange network, GitHub) serves to highlight notable examples that are widely used, discussed and appropriated (as shown by the extensive use made of them by hackers and by the frequent self-reflective discussions about the role that these specific sites have in contemporary hacker learning), and that have successfully developed learning spaces sustained by computation itself.

7.3.1 ”Civilized discourse” as environment for informal learning

Online forums, IRC, mailing lists and direct email exchanges have had a long-standing popularity with developers since the pre-Web era as they allow efficient loose cooperation within distributed communities. They also provide a relatively low barrier to entry for beginners, or for more experienced developers wishing to get acquainted with a new project, as they don't involve complex setups or the need to use specialist or expensive software in order to participate in discussions.

From the point of view of a beginning developer, however, established communication infrastructure used by many free software projects poses in the first place social, rather than technical, challenges: it may be impractical to join a forum just to seek clarification on some past discussions which may have hinted at information a visitor was looking for but may then have drifted off to tangential topics or may have been taken over by trolling or other antisocial behaviour; likewise, there are often few obvious clues that can help a user new to a discussion forum to assess the authoritativeness of participants to discussions. Moreover, as the increased adoption of codes of conduct within developer communities suggests (e.g. Torvalds 2015), discussion-oriented (rather than problem-oriented or question-oriented) forums can often be hostile environments, and the effects of mediated communication on fostering positive and welcoming environments rather than reproducing existing power (im)balances have extensively been questioned, as highlighted in Chapter 2 (e.g. Lea and Spears 1991).

Aiming to address these challenges, the Stack Overflow questions and answers website[55] dedicated to programming questions was launched in 2008 and gained broad popularity in the following years, due its peculiar design goals. Firstly, it presents users with a *problem- or question-oriented interface*: each thread is explicitly meant to contain a space within which a technically sound solution to the problem explained by the original poster is identified, through brief, focused and terse discussion between the individuals involved. The main innovation introduced is the ability for users to grade the quality of the original *question* by ’upvoting’ or ’downvoting’ it according to its originality, clarity and usefulness. The quality of questions is also assessed based on the original poster's ability to formulate context-specific questions, giving examples of troublesome code if this is relevant to the question: this enables users coming to a thread in the future to quickly assess its relevance to their specific research intent, avoiding to read through solutions that may not be applicable anymore to current versions of the same software discussed in the thread.

Secondly, questions eliciting opinionated answers based on personal preferences rather than technical soundness can be flagged by users as off-topic and typically don't attract significant responses. This feature is meant to discourage the meandering conversations often found on discussion-oriented forums, with the aim of limiting opportunities for inconclusive threads or antisocial behaviour.

Moreover, the problem-oriented nature of the forum is enforced through the incentive for original posters to mark one (and only one) of the proposed solutions as 'accepted', thus allowing fast screening of discussions by users reading a thread while researching for a solution to a similar problem. Finally, although the code of the software running the site is proprietary, every contribution made by forum participants is made available under a 'Creative Commons Attribution ShareAlike' license, thus explicitly allowing broad reuse of content posted on the forum.

Social aspects of hacker learning are further tweaked through a site-specific flavour of *gamification*: an incentives system operates throughout most actions that users can take on the site, by adding points to each user's reputation score on the site and by attributing *badges* for specific achievements; users' scores and badges are displayed prominently by the side of key contributions they make to discussions, therefore promoting a view that users' authoritativeness is multifactorial and could to a certain extent be *measured* through quantitative parameters: this is meant to sustain what the site operators see as a framework of civilised, problem oriented discourse by rewarding not only quality of discussion and technical competence but also overall adherence to the site's code of conduct. Users who routinely provide clear and competent solutions to questions, who abide by community rules and enforce them through peer voting of other users' actions can therefore gain a considerable reputation score and visible acknowledgement of their contributions; the multifactorial construction of reputation is algorithmically tuned to try to avoid some of the potential biases of traditional discussion-based forums in which rhetorical ability and

[55] `https://stackoverflow.com/`

influential personality traits may skew discussions and lead community members to favour the opinions of dominant users (including those of users who impose their views through abusive behaviour and by demeaning other forum members), rather than the substantive merit of discussions.

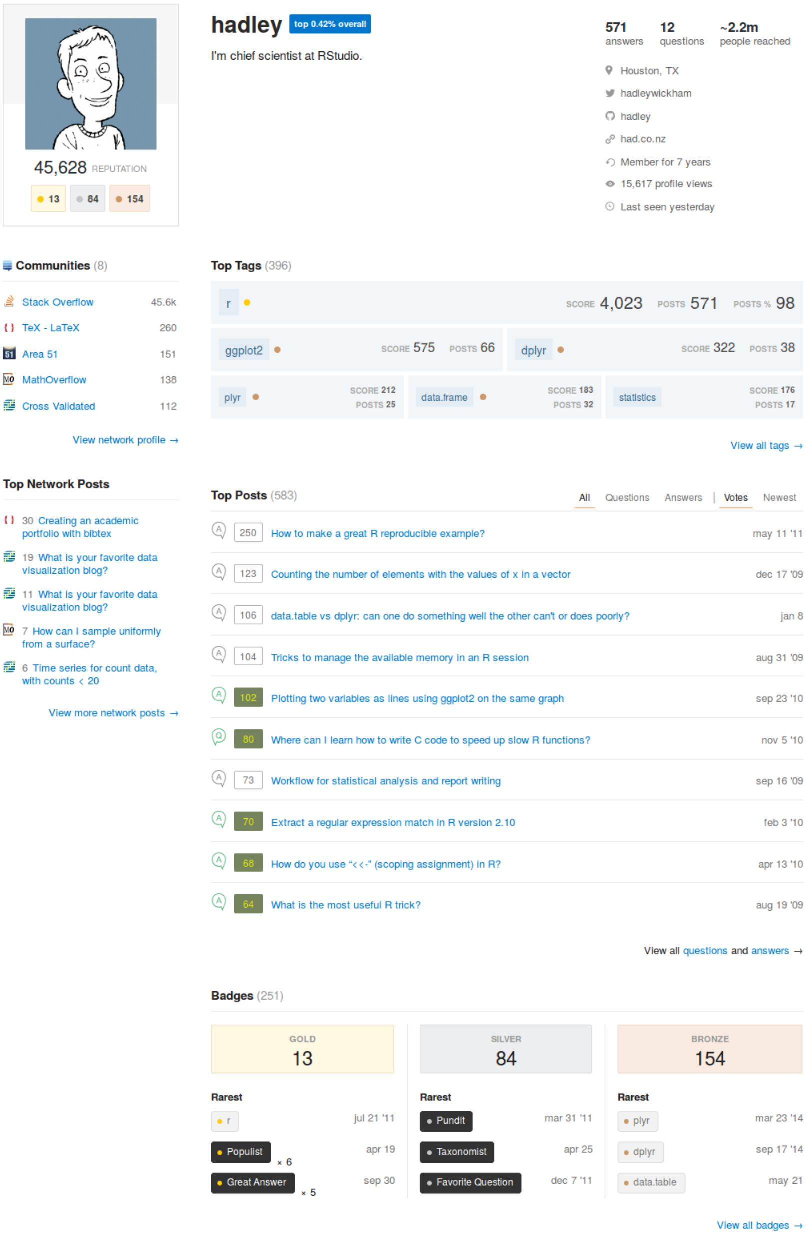


Figure 7.2: User profile on Stack Overflow

User profile on stackoverflow.com for one of the top 0.5% users by reputation (user 'hadley'). User reputation, number of badges earned through the platform's gamification features and an overview of user activity, areas of expertise and main achievements show 'at a glance' the platform's multifactorial view of the user's expertise and contributions.

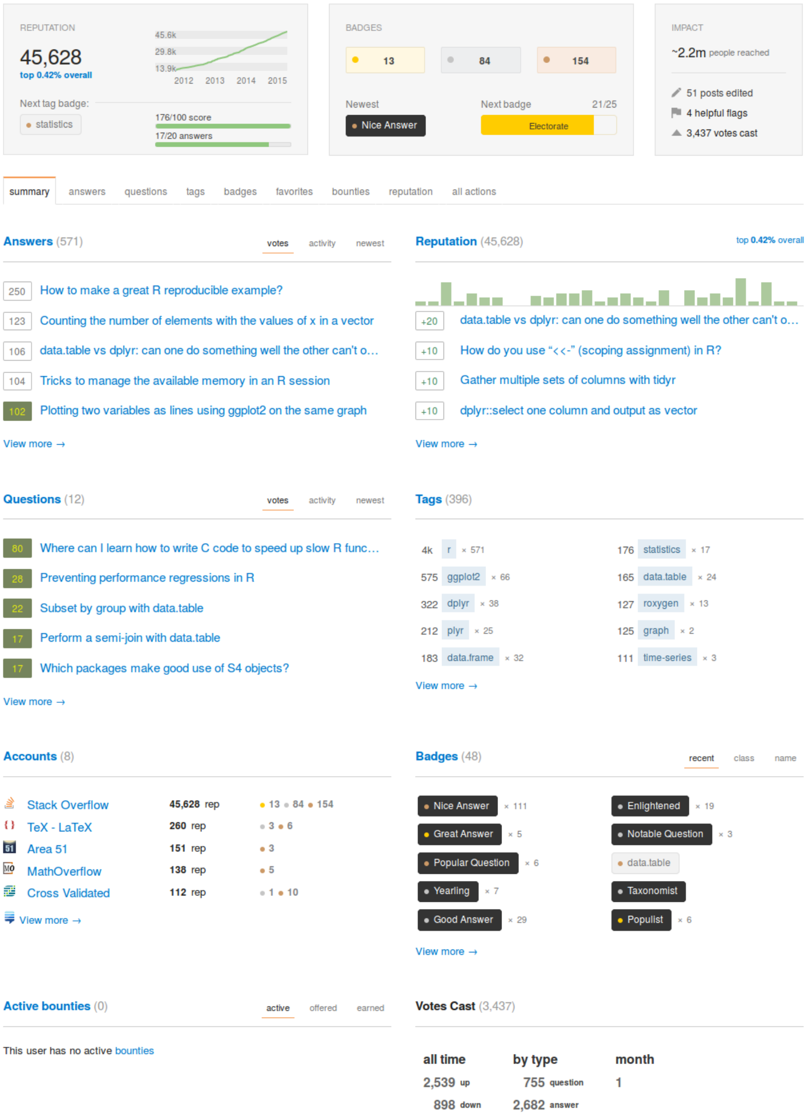


Figure 7.3: User activity summary on Stack Overflow

User activity summary on stackoverflow.com for one of the top 0.5% users by reputation (user 'hadley'). Only the *summary* tab is shown here for brevity, but the visual layout of the page should give a clear sense of the complex array of factors through which an user's activity and contributions are linked to her/his reputation score and earned badges, by way of algorithmic rating of any site-wide activity involving each user.

7.3.2 Computational platforms for hacker learning

As the original Stack Overflow forum quickly established itself as a useful peer-learning resource for programmers[56], the underlying infrastructure and principles were abstracted in 2009 by creating separate forums focused on specific non-programming aspects of computing, further leading in 2011 to the creation of a platform (Stack Exchange) that could support a wide range of different forums, each focused on specific topics, based on the same principles that govern Stack Overflow's social and learning dynamics. Throughout the years, specialized computing-related forums were created (focused, for example, on WordPress development, systems administration, the TeX and LaTeX typesetting engines, etc.), as well as a wide variety of other forums that extend the successful learning model and social environment of Stack Overflow to other technical or scientific domains (e.g. statistics, mathematics) and to everyday life (e.g. cooking, language learning). Interestingly, structured procedures are in place also for the *creation of new forums* on the Stack Exchange platform, enforcing successive stages that allow to assess actual interest and usefulness of a potential new forum in order to avoid opening new spaces for discussion that may end up not attracting enough competent users to sustain useful question and answer exchanges, therefore failing to become valuable learning resources.

Although research targeted at evaluating causal links between Stack Exchange's distinctive structure and the quality of answers constitutes a relatively new research focus, recent studies show contrasting results: on one hand, quantitative analysis of user participation highlights how a core set of users is 'responsible for the bulk of the core activities' (Sinha, Mani, et al. 2013), in ways similar to discussion-oriented forums in major free software projects, leading the authors of the study to posit that '[t]he award process can be further fine-tuned to incentivize content creation activities' (ibid.). A second study (Gkotsis, Liakata, et al. 2015), however, found that the linguistic indicators of the effort made by users while crafting responses to questions is consistently linked to acceptance rates. Further research will be necessary to assess the effectiveness of the social features of Stack Exchange and of its gamification-based incentives system, especially qualitative research aimed at understanding the effects of new types of forums on learning processes: to what extent the Stack Exchange approach, while promoting better quality of answers, translates to easier access to reliable learning resources for beginning developers.

[56] Site popularity data from quantcast.com shows a substantially linear growth of unique site visits—with occasional troughs over summer months—from an average of 100,000 per day in June 2009 to 1 million per day in January 2011 to 5 million per day in November 2015.

Whereas Stack Overflow and Stack Exchange were explicitly engineered to overcome the issues that their creators had identified in incumbent forum platforms such as Experts Exchange[57] ('Stackoverflow is sort of like the anti-experts-exchange (minus the nausea-inducing sleaze and quasi-legal search engine gaming) meets wikipedia meets programming reddit.', Atwood 2008), alternative platforms have also emerged within the same time-frame (e.g. Quora[58], created in 2009), each based on a distinctive set of principles and incentives, and similarly attempting to exploit the 'crowdsourcing' tenet of Web 2.0 in order to build platform value thanks to the unpaid labour of contributors. The aptly named Discourse[59] is a free software forum platform[60] whose aim is to provide infrastructure for self-managed forums based on similar principles as those used across the Stack Exchange network, promoting civilized discussions:

> Today's forum software has a default FAQ that tells you how to make bold text, but absolutely nothing about how to moderate your forum, deal with trolls, appoint moderators, or even how to get people to visit your forum in the first place. The hard-earned sociological lessons of these 10+ year old forum communities are not being baked into their forum software at all. (Discourse project 2013)

Although further qualitative research would be needed in order to better understand causal relationships between the innovations introduced by these Q&A and forum software platforms and users' ability to participate in discussions, to learn from existing answers and to extend and expose their expertise by contributing answers, the success of these alternatives to plain legacy discussion-oriented forums can be inferred at least by the consistently high ranking of Stack Exchange threads within search engines results for Internet development topics, as well by the adoption of the Discourse software by an increasing share of free software projects as their core platform for user and developer discussions.

The relevance of these platforms to Read/Write Internet is twofold: on one hand they disrupted the long-standing pervasiveness of the genre of discussion-oriented online forums, which predates the Internet and, since the era of dial-up BBSes, constituted an ex-

[57] Experts Exchange is an earlier forum platform focused on attracting high quality answers to questions; only questions can be freely viewed, whereas access to answers requires a paid subscription.

[58] `https://quora.com/`.

[59] `https://discourse.org/`.

[60] Founded, amongst others, by Jeff Atwood, one of the creators of the Stack Overflow forum.

tremely popular early form of computer-mediated Read/Write practice. Notably, whereas forums have always been necessarily a Read/Write space, the central disruptive trait of the platforms analyzed in this section consists mainly in allowing users' *write* practices to potentially reach a much broader *read* audience: whereas contributions on discussion-oriented forums can be found mainly through textual signals (full-text indexing of threads and counts of inward references from other websites), structured question-oriented forums provide much richer signaling through reputation scores, acceptance of answers, date of last update, upvotes, etc., allowing quality answers to stand out in search engine results and therefore to be more likely to be found (and potentially, further upvoted[61]).

Secondly, within the context of the development of a new type of hacker public on which this chapter is focused, the increased availability of accessible, peer reviewed and context specific learning resources can empower both beginning and expert developers to quickly find answers to issues they may be facing while developing software, without having to skim through long discussions just to find relevant contributions. Considered together, these changes inform a transition to new configurations of learning and knowledge reproduction processes that are allowing a broader public to be inducted to hacker discourses, technical knowledge and discussions about the social role of computation: whereas this does not necessarily translate to the enrolment of hackers in alternative rationalizations, the point of this analysis is that these reconfigurations are *also* enabling hackers to discuss concerns with mainstream and centralized Internet services, to promote their sense of urgency for alternatives to other hackers, and to develop collaboratively technical knowledge that can sustain actual implementations of alternative rationalizations.

Algorithmically-tuned discussion environments can obviously be criticized for the potential closure of free discussion, which is often as useful for the building of shared representations and to enable developers' collaboration and sense of belonging to a project's

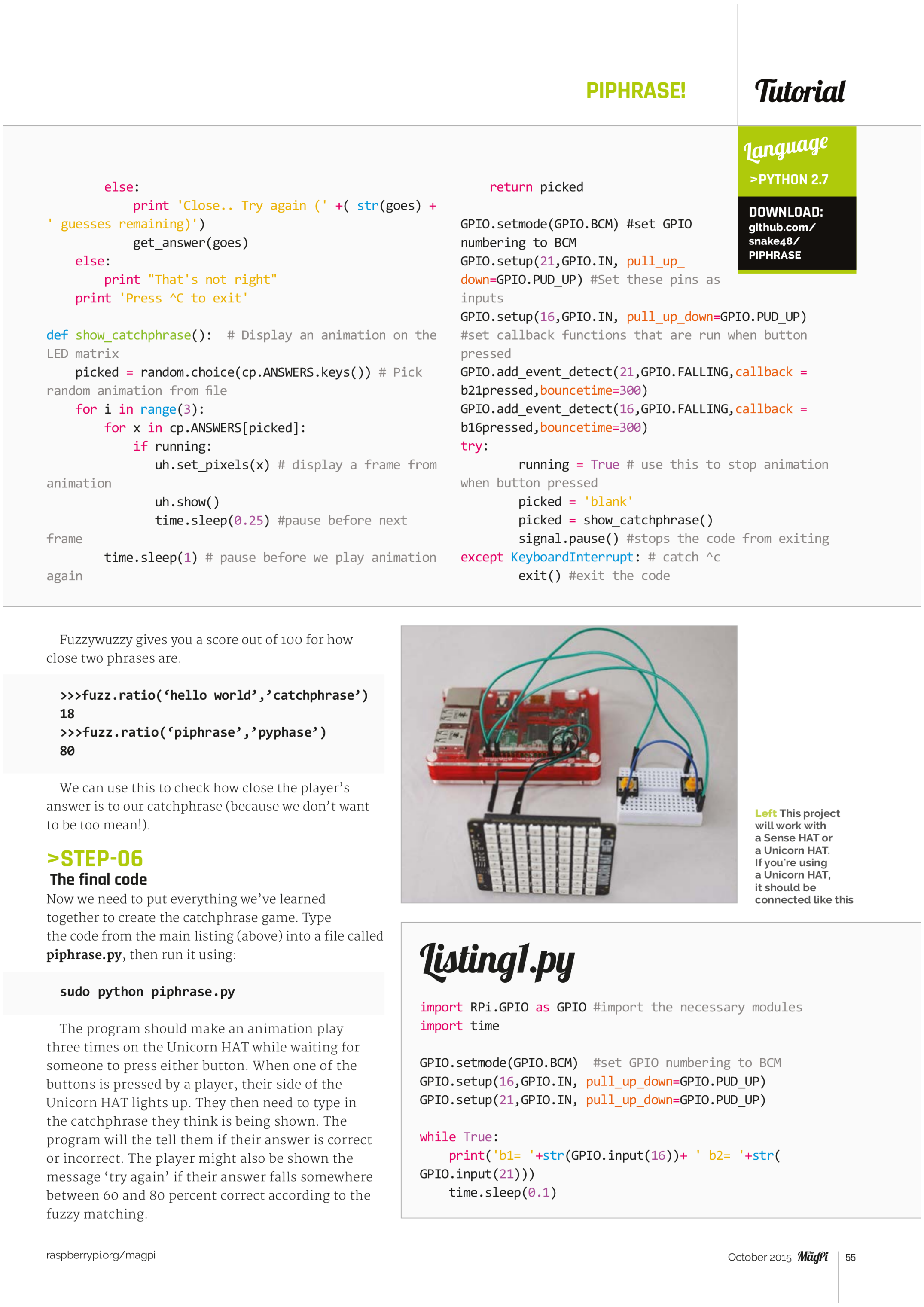
[61] Upvoting a good quality answer is an accepted way to highlight the effectiveness of the solution it provides; however quality answers can become so easily findable that the enthusiastic response to them by users new to the forum is considered unwelcome 'noise': so much so that another algorithmic feature is in place in the Stack Exchange platform, enabling contributors whose reputation score is higher than a set threshold, and under algorithmically enforced conditions, to visibly mark popular answers as 'protected' in order to fence a good answer from surrounding textual pollution typical of discussion-focused forums: 'Some questions are protected because they are expected to attract either spam or users—often new users—who may mistake the site as a traditional forum, posting "noisy" answers such as "Thank you" or "This worked for me" or "I'm also having this problem".' (Meta Stack Exchange Contributors 2014).

core aims and purpose as the outcome of the discussions in itself; however, the success of the Stack Exchange and of the Discourse models have visibly highlighted the potential dysfunctional traits of traditional computer-mediated technical forums (trolling, inconclusive threads, high emotional and social barriers to entry especially for users from underrepresented groups such as women and non-English speaking users), as well as attesting to developers' interest for quality learning resources, thereby creating a higher standard to which discussion-based forums are held.

7.3.3 Taking code for granted: learning to write by reading others' code

At its core, hacker learning ultimately translates to improving one's ability to write software code: this applies especially to developers, but to a lesser extent to designers, product managers and other Internet professionals, who are often required to have some basic proficiency in understanding code in order to be able to collaborate on complex projects.

Similarly to the discursive, text-based aspects of hacker learning discussed above, code-focused learning has been deeply affected by the computational turn of the Internet, enabling new learning strategies and extending learning opportunities for a more diverse hacker public. Learning to program by reading code written by more experienced developers has been a key practice since early stages of computing history, and became widespread with the availability of the first home computers and related hobbyist magazines in the 1970s: these often also worked as analog forms of software distribution, whereby readers would *type* source code from the printed into their computers, learning by example along the way. This very material learning process has recently been the subject of a more mainstream renaissance thanks to magazines for hobbyists associated with cheap microcontroller and computing boards such as the Arduino and the Raspberry Pi, helping readers to assemble simple IoT projects through drawings and photos of component assembly and source code listings: even when source code files are also made available online for convenience, authors of tutorials often recommend that readers type in code in order to slowly progress through its details, which may be lost if just running code downloaded from a website.



PIPHRASE!

Tutorial

```
        else:
            print 'Close.. Try again (' +( str(goes) +
' guesses remaining)')
            get_answer(goes)
    else:
        print "That's not right"
    print 'Press ^C to exit'

def show_catchphrase():  # Display an animation on the
LED matrix
    picked = random.choice(cp.ANSWERS.keys()) # Pick
random animation from file
    for i in range(3):
        for x in cp.ANSWERS[picked]:
            if running:
               uh.set_pixels(x) # display a frame from
animation
               uh.show()
               time.sleep(0.25) #pause before next
frame
        time.sleep(1) # pause before we play animation
again
```

```
    return picked

GPIO.setmode(GPIO.BCM) #set GPIO
numbering to BCM
GPIO.setup(21,GPIO.IN, pull_up_
down=GPIO.PUD_UP) #Set these pins as
inputs
GPIO.setup(16,GPIO.IN, pull_up_down=GPIO.PUD_UP)
#set callback functions that are run when button
pressed
GPIO.add_event_detect(21,GPIO.FALLING,callback =
b21pressed,bouncetime=300)
GPIO.add_event_detect(16,GPIO.FALLING,callback =
b16pressed,bouncetime=300)
try:
        running = True # use this to stop animation
when button pressed
        picked = 'blank'
        picked = show_catchphrase()
        signal.pause() #stops the code from exiting
except KeyboardInterrupt: # catch ^c
        exit() #exit the code
```

Language

>PYTHON 2.7

DOWNLOAD:
github.com/snake48/PIPHRASE

Fuzzywuzzy gives you a score out of 100 for how close two phrases are.

```
>>>fuzz.ratio('hello world','catchphrase')
18
>>>fuzz.ratio('piphrase','pyphase')
80
```

We can use this to check how close the player's answer is to our catchphrase (because we don't want to be too mean!).

>STEP-06

The final code

Now we need to put everything we've learned together to create the catchphrase game. Type the code from the main listing (above) into a file called **piphrase.py**, then run it using:

```
sudo python piphrase.py
```

The program should make an animation play three times on the Unicorn HAT while waiting for someone to press either button. When one of the buttons is pressed by a player, their side of the Unicorn HAT lights up. They then need to type in the catchphrase they think is being shown. The program will the tell them if their answer is correct or incorrect. The player might also be shown the message 'try again' if their answer falls somewhere between 60 and 80 percent correct according to the fuzzy matching.

Left This project will work with a Sense HAT or a Unicorn HAT. If you're using a Unicorn HAT, it should be connected like this

listing1.py

```
import RPi.GPIO as GPIO #import the necessary modules
import time

GPIO.setmode(GPIO.BCM)  #set GPIO numbering to BCM
GPIO.setup(16,GPIO.IN, pull_up_down=GPIO.PUD_UP)
GPIO.setup(21,GPIO.IN, pull_up_down=GPIO.PUD_UP)

while True:
    print('b1= '+str(GPIO.input(16))+ ' b2= '+str(
GPIO.input(21)))
    time.sleep(0.1)
```

raspberrypi.org/magpi

October 2015 MagPi 55

Figure 7.4: Code and electronics assembly instructions from a maker zine
Source code listings and electronics assembly instructions, from MagPi zine, Issue 38 p55, October 2015 (`https://www.raspberrypi.org/magpi/issues/38/` — CC BY-NC-SA 3.0).

Public availability on the Internet of source code has been a long-standing distinctive trait of FLOSS projects; established projects occasionally also make available 'style guides' that help people interested in contributing code to the project to get acquainted with the overall structure of its source code and to make sure that their own contributions fit well with existing code, both in terms of functionality and of code aesthetics. Although most FLOSS projects use similar strategies and tools to host source code and documentation for developers, access to these has traditionally been ad-hoc for each project: developers wishing to study the software or to contribute to it would have to navigate through the project's website, read through documentation on how to submit their contributions, and often discuss these workflows through IRC chat or mailing lists with core developers.

Source code hosting services such as SourceForge[62], providing a collaboration-oriented user experience similar across all the hosted projects, have therefore been a popular option for FLOSS projects wishing to facilitate developer collaboration. The GitHub[63] code hosting service, operating since 2008 and by far the most popular such service amongst developers at the time of writing, has rapidly become a *de facto* default choice for public collaboration on source code[64]. Whereas at face value it may not seem vastly different from other similar services, except for the clean user interface in line with current web design trends, I posit here that its success and popularity derive from the way in which it sustains *computational workflows* and from its role as a learning resource.

At its core, GitHub provides a web interface to source code repositories, to bug tracking and to a wiki area that can be used for project documentation, similarly to most other code hosting services; however, whereas the workflow of older services is typically focused around *projects*, within GitHub the *individual developer* is the atomic social entity around which techno-social interactions are organized. Each developer can create, within the GitHub web app itself, one or more personal copies of any other code repository on

[62] `http://sourceforge.net/`

[63] `https://github.com/`

[64] As of 2015, GitHub was hosting 28 million source repositories with 11.2 million registered users (GitHub 2015), versus 430,000 projects and 'over 3.7 million registered users' (SourceForge 2015) of the second most popular code hosting service, SourceForge, which has been operating since 1999, almost ten years earlier than GitHub. Other services, especially based on the same Git source code management system as GitHub, are preferred by developers who wish to avoid proprietary services, although these may still publish a reference copy of their projects' code to GitHub to facilitate collaboration.

the site: the personal copy (called 'fork') is available instantly and allows the developer to freely work on it without being granted any special permissions to modify the canonical copy of the project they are collaborating on; if she then wishes to submit her own improvements, she can do this through a 'pull request' (a request to merge the improvements into the canonical copy of the project), which can be accompanied by comments and can be discussed directly through the GitHub web app.

This organization of collaboration workflows around each individual developer is not a unique feature of GitHub but rather a *modus operandi* enabled and promoted by the Git source code management system[65], which was the first such system to provide a widely accessible peer to peer interface to code collaboration, removing the need to maintain a canonical central copy of the source code and allowing contributions to be coordinated directly between developers with no central authority. The GitHub service, however, successfully developed a web user interface that wraps aspects of the computational complexity of the processes involved in managing collaboration on source code, up to removing the need to use any specialized software on a developer's workstation for a range of simple changes that may be done entirely through GitHub's web interface, similarly to the software affordances provided by popular social network sites, matching common patterns of social interaction online through algorithmically structured software counterparts. A parallel between SNSes and GitHub-like 'social coding' in fact extends beyond the encapsulation of procedures (whether social or technical) within software code: Dabbish et al. (2012) identify recurring patterns of complex social interactions happening within the GitHub platform whereby developers interpret the rich metadata exposed by GitHub about activity on code repositories as signals related to interests, expertise, intention and competence of other developers; moreover, although further research would be needed to develop a deeper understanding of the implications of individual-centric — rather than project-centric — developer collaboration workflows, my own research in this field (with Pozzi: rota and Pozzi 2013, unpublished working draft, a quantitative analysis of a large dataset including all the activity between 2011 and 2012 across publicly accessible repositories on GitHub) interestingly highlights how this type of workflow is nevertheless successfully appropriated by developers to create highly collaborative interactions as well as exchanges of knowledge around popular FLOSS projects, rather than leading to more isolated coding practices.

By allowing other web-based apps for developers to integrate with its infrastructure, GitHub effectively provides a sort of 'computational workplace' for developers: external services that are designed to integrate with projects hosted on GitHub, for example, provide the ability to automatically test any new source code published[66] or to manage project tasks, resources and schedules, or allow developers to chat in real time, keeping logs of conversations for future reference, and enabling to add to conversations non–human 'bots' ready to interpret instructions given to them and to act accordingly—for example, by creating a new 'issue' (a bug report) in the project's issue tracking system, based on a problem discussed in a developer chat room. Whereas developers working in large companies may typically use corporate infrastructure for these tasks, their availability as free or cheap services is an enabling factor for individual developers or small groups of volunteers lacking the resources to set up and maintain complex code collaboration infrastructures.

Informal learning through the ability to read other developers' code is enabled by the visibility of software code on GitHub: on legacy source code hosting services the web interface to each project is often organized around a textual presentation of the project, whereas the web interface to source code is a distinct web application accessible through layers of navigation menus. On GitHub, instead, the 'source code view' is the default and main way to browse a project; alongside it, a 'README' file is displayed, if available, and developers are invited to provide a clear and thorough overview of their code in it, in order to help clarify what a software program does and how it works, by combining this technical description and the act or browsing through the source code:

> Consider the process of writing the Readme for your project as the true act of creation. This is where all your brilliant ideas should be expressed. This document should stand on its own as a testament to your creativity and expressiveness. The Readme should be the single most important document in your codebase; writing it first is the proper thing to do. (Preston-Werner 2010)

[65] `http://git-scm.com/`.

[66] This (computation-based) practice, known as 'Continuous Integration', is increasingly widespread especially amongst large distributed software projects, as it helps to make sure that any new code works correctly and that it doesn't cause existing functionality to break, by introducing 'regressions', which could be very difficult to notice without a computational testing infrastructure.

This seemingly superficial change in focus — from narrative about a software's *functionality* to a technical summary of its *implementation details* — could be read, as most of the other changes discussed in the first part of this chapter, as signaling an audience shift: whereas projects producing 'desktop' software needed to convince potential (and potentially non-technically inclined) users to try their applications, this marketing function is now mostly delegated to app stores[67], whereas the code repository on the web — whether on GitHub or elsewhere — can be structured around the aim of attracting developer collaboration.

The code- and developer- oriented cognitive and social structure of GitHub has become dominant and pervasive in developer discourses in the short timespan since the service's creation: similar popular services essentially mimic GitHub's dynamics, and in some case the actual visual layout of information and user experience. The resulting primacy of source code visibility has contributed — according to several developers — to a shift in attitude towards software code published online: first observed by Governor (2012), who labelled it as 'Post Open Source Software' (POSS) in a tweet ('younger devs today are about POSS—Post open source software. fuck the license and governance, just commit to github'), POSS synthesizes the pragmatic attitude of an increasing number of software developers who don't provide explicit licensing information for the software they publish on GitHub, as if the act of sharing their code on the site was to be interpreted as a public domain dedication, and equally assuming that any other code without an explicit license published by others on the same website is to be considered free to use, whereas in most legal contexts worldwide the absence of explicit licensing information implies an 'all rights reserved' copyright. Although others (Berkholz 2013; Fontana 2013; Villa 2013) highlight the troublesome legal implications of assuming freedom to reuse software code that is not clearly licensed, the Post Open Source Software phenomenon highlights how visible, ubiquitous and central source code has become in the learning processes of Read/Write Internet hackers.

[67] GNU/Linux distributions have traditionally provided a similar mechanism through extensive and ever-growing directories of installable FLOSS 'packages' as integral part of the each distribution.

7.3.4 Beyond visibility of code: learning computational thinking

The ability to learn from others' code — and for experienced developers to improve their own command of technologies and methods by informally mentoring junior developers — is further fostered by other subtle affordances introduced by GitHub, leveraging the features of the underlying Git source code management software: from the ability to reference *specific portions* of a source code file (rather than only the file itself), to the possibility to observe each source code file's changes through time, to checking which developer introduced specific changes, and to referencing any discussions about features or bugs affecting specific portions of source code.

Web-based live-feedback coding environments, increasingly popular since 2008 when the first such project (Js Bin[68]) was created by developer Remy Sharp, are a further genre of tools focused on visibility of software code in hacker learning, with the specific aim of sustaining *computational thinking*. As part of a relatively recent genre subject to wide experimentation, these environments leverage the computational capacity of modern web browsers[69] in variously configured attempts to address the long-standing and increasingly relevant issue of managing complexity in software (Brooks 1987; Moseley and Marks 2006), promoting *visualization* of *code execution* over plain *visibility* of *code itself*. Whereas common 'pastebin' web applications enable occasional and informal sharing of text anonymously online, live coding environments computationally extend this genre by adding the ability to share short JavaScript programs and to run these to get immediate feedback on changes applied to the source code without any need to set up a test website or to use specialized software other than a modern web browser. By providing a quick and free way to share *runnable* code examples, web developers use them to complement developer-focused code sharing services and the problem-oriented forums outlined earlier in this chapter: it is commonplace to see runnable code examples referenced in Stack Overflow discussions, and often the example given by the person asking a question is then modified by respondents and shared back in its final state, after any issues have been fixed, as part of the answer given.

---

[68] `https://jsbin.com/`

[69] See Chapter 4, section 'The *computational turn* of the Internet'.

More importantly than providing a convenient way to share runnable examples, such live coding environments enable a computational pedagogic workflow for hacker learning, whereby source code and the effects of running it are visible one by the other, allowing learners to experiment by changing individual portions of code and inferring the effects of these changes through immediate visual feedback. These kinds of live coding environments have become a core component of current informal and formal (e.g. websites providing structured web development courses) learning resources, with several of these (e.g. Heilmann 2012; Resig 2012) referring to a popular conference keynote and successive blog post by Bret Victor (Victor 2012a,b, respectively) as source of inspiration through his concept of 'Learnable programming', made possible by the computational environment in which coding experiments can take place:

> In an environment that is truly responsive you can completely change the model of how a student learns: rather than following the typical write -> compile -> guess it works -> run tests to see if it worked model you can now immediately see the result and intuit how underlying systems inherently work without ever following an explicit explanation. When code is so interactive, and the actual process of interacting with code is in-and-of-itself a learning process, it becomes very important to put code front-and-center to the learning experience. (Resig 2012)

Victor, however, distanced himself (Victor 2012b) from endorsements of his work as inspiration for live coding environments: on one hand he acknowledges (by citing in turn Papert's seminal work in this field: Papert 1980) that direct manipulation of code and visualization of the effects of such manipulations is one of the learning strategies essential for the understanding of the normally invisible links between code execution and its outcomes (Berry 2014, chapter 4):

> **People understand what they can see.** If a programmer cannot see what a program is doing, she can't understand it. (Victor 2012b, emphasis in the original text)

However, he also stresses that an equally vital aspect of developer learning requires a more complex and principled configuration of the learning environment, exposing — rather than hiding — the computational thinking that lies behind the visible affordances of programming language features:

> **Programming is a way of thinking, not a rote skill.** Learning about "for"

> loops is not learning to program, any more than learning about pencils is learning to draw. (Victor 2012b, emphasis in the original text)

The 'learnable' coding environment that Victor proposes is, accordingly, structured around the need to visualize 'what the code is *doing*' (ibid.) and how its operation relates to underlying data being processed, rather than simply providing immediate feedback on the final result of code execution. Albeit still largely experimental, the genre of live coding environments exposing the computational complexity of software code is an increasingly relevant learning toolset for hackers of Read/Write Internet, gathering extended interest in online hacker discourse and progressively enabling developers to focus less on the 'rote skills' of software development and more on the imaginative uses of software, among which are those analyzed in detail in the next chapter.

7.4 A recursive public of hackers

Building, maintaining and operating extensive Internet infrastructures able to serve the needs of increasingly large user bases is a complex task requiring vast resources and significant capital investment that are beyond the reach of independent developers and small companies with no access to venture capital[70]; ultimately, however, all software and hardware being used at some layer of the Internet is designed and built by developers, engineers and other Internet professionals: even when these are employed by companies and have to act according to corporate directives, the ways in which software and hardware are produced include a multitude of material factors (expertise, technical opinions, awareness of possible alternative ways to undertake certain tasks, availability of reusable software libraries each embedding other developers' choices and opinions, etc.) which, given a defined product design, can lead to different outcomes and ultimately to different ways in which web apps and internet devices are used and attain success amongst end users.

Freedom of developers and engineers working within corporate settings should certainly not be overstated, as modern software development practices can be repetitive and alienating as other kinds of work (Berry 2014, p63, and as also memorably popularized by

[70]Either by choice—because of ethical or strategic reasons—or because their projects are not considered a good fit by venture capitalists.

Coupland 2007); similarly, even though programmers are the ones ultimately writing, testing and shipping code, modern software engineering practices help ensure that plans and aims set by management are suitably complied with. Developers working independently, either by volunteering their expertise to free software projects or by working on their own products with no obligations to respond to external capital or managerial pressures, may enjoy more latitude in their choices, even in commercial settings where the ultimate goal is to achieve financial viability and success.

Whether acting in a professional or personal capacity (or various combinations of the two, as is often the case depending on individual circumstances), Internet developers constitute a peculiarly public relevant social group: they contribute to building the Internet and at the same time use the Internet as the main channel through which their practices, motivations and narratives are discussed. It is useful here to employ Kelty's framing of *recursive publics* (2008):

> I call such publics *recursive* for two reasons: first, in order to signal that this kind of public includes the activities of making, maintaining, and modifying software and networks, as well as the more conventional discourse that is thereby enabled; and second, in order to suggest the recursive 'depth' of the public, the series of technical and legal layers—from applications to protocols to the physical infrastructures of waves and wires—that are the subject of this making, maintaining, and modifying. (Kelty 2008, p29)

Whereas Kelty's focus is specifically on 'geeks' involved in the development of free software projects, the formal structures he describes as constitutive of a recursive public can also successfully capture a broader context of recursive practices and connected actors that in the recent years on which the present chapter is focused (post-2010 or post-computational turn of the Internet) apply to an increasingly large proportion of Internet developers: the free software developers followed by Kelty constituted an highly technically competent vanguard of developers who were coordinating their geographically distributed collaboration through the same software infrastructure that they were building, whereas the recent evolution of hacker learning and software engineering practices discussed in the previous sections has contributed to attracting less pioneering developers (and, often, people without any former extended background in software development) to Internet and mobile development.

Kelty's concept of recursive publics is the (circular, or recursive) discursive framing that more closely reflects the peculiarities of the 'hackers' of this chapter. On one hand, the publicity of their actions and opinions (through words and published source code[71]) sets them apart from the faceless labour of developers who may show up at work, write code addressing goals set by managers and go back home in the evening, without leaving a public trace of their actions, motivations and concerns: this publicity contributes to the building of a global hacker discourse which both defines what it means to be, today, a developer with interests and concerns beyond the technical details of daily work, and functions as very visible guidance for developers wishing to take part in this global hacker discourse, and therefore as a way for the discourse itself to be socially reproduced. Hackers' keen interest for the reproduction of hacker discourses is indeed a distinguishing trait and reason for the choice of the 'hacker' term in this chapter.

Moreover, other traits variously discussed in previous literature (Alleyne 2011; Coleman 2012a,b, 2014; Jordan 2008; Kelty 2008; Levy 2001) apply to the developers on which this chapter and the next one are focused: a fascination for intellectual challenges, a deep understanding of strategies and tools that could help addressing these, as well as the passion and resolve to pursue them. Most of the accounts of developers mentioned here show, accordingly, the ability to reframe incumbent paradigms in ways that allow alternative interpretations and the repurposing of available tools to develop creative solutions (i.e. by way of 'hacks'). Even when this is done as part of daily salaried work, the public discussions around it often highlight the attractiveness of challenges for the sake of intellectual curiosity; this is often manifest in the popularity of 'side projects' that channel interests, methods and tools developed throughout daily work into creative endeavours pursued in the evenings and at weekends, often becoming informal, self-directed and continuous training that occasionally is turned into a new business and that is considered a distinctive trait of hacker spirit, even granting monthly column space on technical magazines for Internet professionals.

[71] As mentioned earlier in this chapter, either publishing code after careful curation for others to use or just 'dumping' it on github.com or elsewhere online (as 'Post-Open Source Software') is an increasingly widespread practice amongst current hackers. The significance of this practice is both cultural, by contributing to the 'hacker discourse' beyond narrative discussions, and pedagogical, by allowing others to learn by example and to provide feedback and criticism. The political significance of sharing code at the heart of Free Software is, however, often less clear throughout more spontaneous practices of publishing source code online.

As I reviewed hacker discourses in the past five years, the self-reflective trait of hackers as recursive public started becoming visible through the proliferation of venues through which non-technical topics were increasingly intertwined with technical ones: most visibly, talks at technical conferences focusing on social issues (e.g. use of Internet technologies for socially progressive projects, surveillance, privacy), on the role and responsibilities of hackers as computational mediators and on 'meta-issues': social issues within hacker communities themselves (foremost, the problems stemming from the hegemonic role of white, middle class, heterosexual male developers). Alongside these, printed and online publications (e.g. Offscreen[72], The Manual[73]—officially focusing on design but often venturing into recursive reflections—One Two One Two Microphone Check[74], The Pastry Box Project[75], etc.) explicitly aiming at developing an understanding of the role of hackers in societies increasingly dependent on computation have become established references for the recursive development of self awareness, as one of such developers interviewed in Offscreen magazine points out:

> I believe and hope that over the next decade, we'll experience a bigger shift in the perception of what web development is or can be: not just a solution for *technical* problems, but one for some of the most pressing social injustices of our time. It's a shift that's already starting to happen. (Joshi 2015)

The processes through which hackers have been collectively building spaces for discussion, learning and self-reflection — and through these, building themselves as a new recursive public able to interpret the present, its own role within it and to act according to personal values — is summarized by the following quote from the final article of the Pastry Box Project, which gathered hacker voices for a similar period of time as my own analysis of hacker discourses (2012-2015):

> The concept at the origin of the Pastry Box is quite simple: bring people together and let them write about anything they want. If you do that, you should get some kind of testimony about our day and age. You should be able to grab some sense of our era. [...] [W]hen I read the texts published in 2012, I can see that the preoccupations of our writers were not the same as the preoccupations of the people writing in 2015, and that 2012 is in many ways a statement of what the web—and our world—was at the time. (Duloz 2015)

7.4.1 Peer-induction to hacker discourses

Software development — and Internet development in particular — has been gathering increasing attention from mainstream media (Andreessen 2011), and is often portrayed as a potentially highly lucrative endeavour associated to fashionable lifestyles—not without occasional 'bad celebrity' (Andy 2013; Bearman 2015a,b) portraits—and through which developers are seen as forming a new type of capitalist elite, building and controlling in practice essential infrastructure of contemporary life. Whereas the largely fashionable images and career prospects thus associated to software development in mainstream media may be a contributing factor to the increased popularity of formal Internet-related learning curricula as well as informal learning opportunities (a proliferation of workshops, hackathons, unconferences and physical hackerspaces is particularly visible in global cities with large software industries such as London, Paris, New York, San Francisco), access to highly paid developer positions is very elusive; however, easier access to learning and very low initial investment required to start experimenting with web and mobile development, career options as Internet developers are attractive and often accessible to determined professionals even with little or no previous training in software development, making the public of Internet professionals of recent years as a whole a much more heterogeneous 'public' than Kelty's geeks, inevitably lacking the moral cohesion of free software developers.

Through the peculiar ways in which Internet professionals who engage publicly in discussions about their own role expose themselves in the public sphere, however, Kelty's core tenet of a recursive public ('A recursive public is a public that is constituted by a shared concern for maintaining the means of association through which they come together as a public.', Kelty 2008, p28) can also meaningfully apply to Internet developers at large. An overview of the public forums where these professionals develop and follow discussions beyond the technical realm highlights, perhaps unsurprisingly, close connections within a globally distributed public: popular forums such as Reddit[76] and Hacker News[77], where topics and news items suggested by participants can be 'upvoted' and 'downvoted' accord-

[72] `http://offscreenmag.com/`
[73] `https://themanual.org/`
[74] `http://one-two-one-two-microphone-check.com/`
[75] `https://the-pastry-box-project.net/`

ing to shared perceived relevance and usefulness, represent a informal 'daily frontpages' of the (English-speaking) development world, shaping the participants' perception (as well as that of casual visitors or of beginning developers) of which topics currently constitute trends, concerns and connections worth pursuing for an Internet developer.

By virtue of its association with the Silicon Valley startup accelerator Y Combinator, Hacker News reflects the technical and entrepreneurial interests of startup founders and Internet professionals close to the startup business model; a review of popular topics[78] highlights concerns that extend beyond technical and commercial to include science, medicine, education and occasional rare curiosities that posters consider likely to elicit fellow forum participants' interests.

The overarching theme within non-technical discussions is often an interest for intellectual challenges in different fields of human endeavour relevant to contemporary life, reflecting an assumed 'hacker spirit' characterized by a fascination for complex and attention-worthy issues, whether abstract (as mathematical theorems) or practical (such as improvement of health or living conditions within specific groups or in specific locations), combined with the belief of being able to contribute to the advancement of these issues through science and technology.

Although mainly appealing within a narrow context of Silicon Valley-style startup entrepreneurship, discussions on Hacker News are nevertheless indicative of contemporary traits and concerns of a highly visible global hacker community: initial research on content quality and discussion bias on social news aggregators such as Reddit and Hacker News (Stoddard 2015) identified commitment to quality curatorial efforts by the forum participants, highlighting how these forums function both as discursive platforms where hacker narratives are identified and fostered, as well as an induction to hacker discourses and mindsets for newcomers. This high visibility of the community and of its discussion forum also results in public exposure of individual hackers whose names are mentioned routinely: from Y Combinator partners, whose blog posts are immediately discussed on the

[76] `https://reddit.com/`

[77] `https://news.ycombinator.com/`

[78] A selection of the most popular articles is published monthly in the independently curated printed zine 'Hacker Monthly' (`http://hackermonthly.com/`).

forum, to popular developers, whose ideas, contributions, occasional rants and sometimes lifestyles are almost religiously followed as exemplar of a specific flair of hacker spirit.

The apparent widening of the domain of 'hacker spirit' throughout this chapter can be considered debatable. Hacker scenes such as those associated to Hacker News and the Silicon Valley startup culture can rightly be critiqued as articulating a largely instrumental view of technological rationality (Feenberg 2002, pp5–6): discussions on Hacker News often revolve around progressive concerns but they also visibly display a widespread interest in capitalist business models based on technological responses to real-world challenges that are assimilable to the strategies labelled as 'solutionism' by Morozov (2013b). Despite the substantive ideological distance between these hacker scenes and those of established hacker subcultures such as the cypherpunk movement, numerous FLOSS projects (for example the Debian project[79]) and the Chaos Computer Club, however, my point through this chapter is that the recursive traits of the strategies discussed in the previous sections highlight a common dedication to reconfiguring technology itself, through a combination of discursive strategies and computational strategies, with the double aim of creating computational capacity through which the needs of users and developers can be addressed outside of hegemonic agendas, and of establishing and increasing the social capacity for reproduction of the hacker public itself. Unavoidably, different agendas shape hacker actions in different ways, leading to different agential outcomes and inscribing different values, capacities and limitations in affordances that are then used by hackers themselves and by common users. In some cases, technical form produced by hacker labour may be certainly concerned with 'making people click ads', to relate back to the opening quote of this chapter, and with responding to capitalist agendas by creating software that controls users (for example, to extract profiling data or through the use of behavioural advertising); at the same time, however, the constitution of hackers as a visible recursive public able to act as computational mediators between the social and the technical is the core shift that makes it possible for alternative rationalizations to be developed by exploiting the same traits of computational capacity that are successfully sustaining mainstream Internet infrastructure and services.

[79]Nevertheless, besides the shared dedication to software freedom, the Debian project accommodates many, often contrasting, ideologies and practices.

Chapter 8

Computational Read/Write Internet: assembling alternative rationalizations

> The web itself is antifragile. It interprets our business models as damage and routes around them. If we've learned, we'll respect this next time we make something. (Marks 2012)

8.1 Introduction

In this final empirical chapter I look at actual alternative rationalizations through which hackers shape affordances that enable users to exert computational agency within lifeworld Internet outside of the limited options provided by mainstream Internet affordances. Among a wide array of alternative rationalizations, I chose to focus on two domains that are specifically related to the limitations and constraints that emerged through the analysis of my local fieldwork, as experienced by myself as a Web 2.0 developer and through the accounts of my research participants: the core concerns of these domains of alternative rationalizations are, respectively, control over one's own personal content and avoidance of centralization of control over Internet infrastructure, which in turn conditions users' ability to exert computational agency outside of the tightly guarded boundaries of corporate walled gardens and information silos. Therefore, alternative rationalizations are here analyzed not just as alternative implementations of mainstream affordances but as attempts to redress power imbalances inscribed in mainstream centralized configurations, through the role of the recursive public of hackers (discussed in the previous chapter) as cultural translators and computational mediators.

Whereas the relevance of the practices and strategies analyzed here is reconnected to my local fieldwork, the details of analysis are concerned with hacker discourses as emerging from publicly available discussions; the empirical materials used throughout this chapter are the same informing the previous one: recordings of presentations at technical conferences, discussion forums, technical blogs, publicly available source code; whereas these were analyzed in the previous chapter with the aim of identifying strategies through which hackers establish themselves as a visible recursive public of computational mediators, the same materials are analyzed here to examine the specific practices through which hackers assemble the technical form of alternative rationalizations. A detailed overview of my source materials is provided in Appendix B.

This chapter is composed of three sections. In the *first section* I analyze the domain of practices related to self-hosting (and controlling/owning) one's own content on the Internet — rather than having to rely on walled gardens and information silos controlled by corporate entities and often provided through proprietary software. Being able to meaningfully control one's own content was the core tenet of the design of my Web 2.0 application whose development struggles have been discussed in Chapter 5, and similarly the accounts of my research participants highlighted their practices related to meaningfully appropriating Internet affordances while dealing with personal content. The *second section* deals with the domain of tensions between the centralization trends of mainstream Internet discussed in Chapter 4 and the 'redecentralization' efforts of hackers concerned with the imbalances of power and control over the lowest, infrastructural fabric of lifeworld Internet that stem from centralization of computation and infrastructure. The *third section* concludes my overview of alternative rationalizations by examining the struggles involved in operating outside of the boundaries of mainstream narratives and technologies:

how these alternatives are represented as relevant to common, non technically-minded, end users and how issues (such as management of identity and trust) that are efficiently managed through centralized configurations are addressed within decentralized alternatives.

8.2 Self-hosting: the IndieWeb

> The independent web takes care of itself because it fulfills basic human desires. The desire to connect. The desire to produce. The desire to enjoy content not mediated by corporations that can't afford to produce anything less than a blockbuster. On the non-commercial web, an audience of 10,000 souls (or even a thousand) is meaningful. (Jeffrey Zeldman, quoted in Kaufman 2002)

8.2.1 Owning one's own content on the web

The Web 2.0 discourse of the mid-2000s contributed to focusing attention on creation and on publishing of content through specialised web applications whose main distinctive traits included the ability for relatively inexperienced users to share content online, or just to store it privately in specialised web applications for personal use.

Although keen Internet users had been sharing personal content online for several years, for example by handcrafting early personal webpages uploaded to popular hosting services such as GeoCities, early popular Web 2.0 apps gained popularity as they succeeded in enabling users without specialist technical skills to create and publish content without having to first become familiar with the details of HTML markup, CSS styling, file formats, metadata for media files, and so on.

As Petersen (2008) highlights, however, the 'user friendliness' of Web 2.0 apps such as Flickr, YouTube and the many blogging platforms that became popular amongst a general public since around 2005 is often part of a more complex set of features and design goals aimed both at making it easier for users to upload content as well as making it hard for them to stop sharing content later on and moving all their personal content to a competing platform. As users' attention and personal data are often the main assets that companies providing user services hold, it is understandable that user experiences are often optimised towards the dual goal of enabling ingestion of content while discouraging (or outright disabling) its export.

Other perceived limitations of proprietary Web 2.0 services, so commonly discussed in mainstream press that they are increasingly part of a shared social imaginary of the Internet (Taylor 2004), include the implicit handing over of content to the platform owners for commercial exploitation (Jenkins 2008), the uncomfortable tradeoff between convenience and privacy (Turow, Hennessy, et al. 2015), as well as more technical reasons such as the lack of control over the ways in which user data is stored, used and preserved; to give a sense of the extent of this phenomenon, the Archive Time collective[80], a team of volunteers whose aim is to archive snapshots of data from websites focused on user-generated content when these are about to be shut down, maintains a list of such website closures ('deathwatch') listing, as of late 2015, over 400 of them since 2001.

Acknowledging such limitations while trying to leverage more recent developments of web services that made it easier for non-technically minded users concerned about privacy and surveillance to publish content online without having to rely on external services, a couple of web developers (Çelik and Parecki) started collaborating in 2010 on the *IndieWeb* movement (Çelik 2015), soon joined by other developers at the first IndieWebCamp workshop in Portland in 2011, an informal gathering (later organized yearly in several worldwide locations) whose aim was to develop IndieWeb tools and to let participants help each other at setting up their IndieWeb personal sites.

Çelik and Parecki's stated goals through the IndieWeb are to enable users to own their data ('owning what you author by publishing it first on your own site, and only later copying to silos', ibid.) and to *selfdogfood*[81] ('using your creations on your own website that represents your primary self', ibid.): a user's self-hosted website and personal domain name become the central repository of a user's own data and expose it both to human visitors (who would be browsing website pages) *and* machines — web applications that replicate copies of user content from their own website to third party web apps (such as photo sharing, calendar or social bookmarking apps), or — conversely — republish to an user's own

[80] https://archiveteam.org/

website their content first published elsewhere (for example on social network sites).

User *creations*, in Çelik and Pareki's statement above, are meant to be both *content* such as blog posts, photos or other artifacts, as well as *software code* that implements IndieWeb strategies and processes on one's own website. This articulation beyond the mainstream content-focused conceptualization of personal content is central to the ethos of IndieWeb, as well as positioning it firmly as a computational Internet phenomenon, as is discussed in more detail in the following sections.

8.2.2 Reading and Writing on the web: IndieWeb strategies

In practice, the IndieWeb's core aims are addressed through multiple strategies, in constant development through the discussions and development efforts of participants in the sparse IndieWeb community. The core strategies suggested to help users retain ownership of their own data revolve around the need to avoid depending on third party services — whether commercial or free, proprietary or open — to publish personal content on the web and typically include:

- the use of a domain name registered on one's own name to host any personal web content
- hosting any personal web content on one's own website, as opposed to using specialised third-party web apps

Self-hosting one's website by using a free software CMS installed on a personal web server (as opposed to using a free or commercial platform such as WordPress.com) is recommended, but in practice this is just a technical detail, as the first requirement listed above (using one's own domain name) allows to replace the underlying software or publishing platform whilst keeping all existing content reachable through the same web address, as long as the software originally used to manage content allows to export content freely.

In practice, then, relying on free software web applications and content management systems hosted on an external platform constitutes a common arrangement for IndieWeb

[81] *Eating your own dog food* or *dogfooding* is a common expression in software engineering circles, referring to the challenge of using one company's own products throughout daily work practices, in order to identify opportunities for their improvement, or, in some cases, as a marketing initiative aimed at highlighting that a certain software is considered so reliable that their own creators can successfully depend on it.

users, as an intermediate configuration between the two extremes of self-hosting web apps on one's own server(s) (and therefore retaining full control over personal data) and using proprietary platforms (and therefore relying on the platform operators' promises about ongoing ability to access and export personal data): although operators of external platforms based on a free software core (such as WordPress.com in the example above) could still potentially discontinue services at short notice or remove features allowing content to be exported, users can normally use these features while they are available and — most importantly — knowing that the most recent 'export' or backup of content will be reusable as-is after importing it into an equivalent service based on the same free software core, be that self-hosted or provided as a service by a competing service provider.

Although the ultimate goal of IndieWeb is to support users to publish their own content and engage in discussions online, one of the distinguishing traits of IndieWeb is that the tools and strategies to support personal publishing are consciously embraced, maintained and curated by each user rather than simply chosen from an array of possible alternatives that could be used without some degree of understanding of what their features are, how they work and how they protect users' ownership of content (or fail to do so).

Within the IndieWeb discourse, each user is considered the social unit around which web content is assembled: in this, the IndieWeb differentiates itself from other counter-narratives of Read/Write Internet such as federated social network sites, which may reject corporate-controlled silos and embrace self-governed tools supporting *communities* of independent users. The centrality of the *individual*, instead, is a core concern of IndieWeb: so much so that there are no set rules or exact requirements to be met in order for a person's web assets to be considered IndieWeb; rather, its proponents explicitly strive to avoid binding IndieWeb to specific technical solutions, with the aim of fostering independence of users not only from corporate Internet silos but also from any other technical agent — even other independent developers or fellow IndieWeb users, and — ultimately — of encouraging users to experiment, trying to structure their web strategies around their own preferences rather than having to model their practices according to features and limitations of tools made available by others.

This is evident both in the brief notes on the origin of IndieWeb (Çelik 2015) and in the IndieWeb project's rejection of *monoculture*. Çelik and Parecki started the IndieWeb movement after attending the first *Federated Social Web Summit* in 2010[82] and realising that the idea of federated web services, albeit a radical improvement over the fragmentation imposed by the increasingly popular corporate silos (Facebook, Twitter, Google+), was nevertheless still constraining users' practices and ability to experiment by inviting them to use 'one-size-fits-all' social web applications (Çelik 2015).

The IndieWeb movement, therefore, while keeping open friendly collaborations with federated social web projects, explicitly rejects what is labelled as *monoculture*:

> *Monoculture* refers to the antipattern of one piece of software dominating (or trying to dominate) its field, often by being limited to communicating with other instances of the same codebase. A monoculture (same software running on servers run by different people) is one step above a silo (same software running on servers run by the same people or organization). (IndieWebCamp 2014a)

This rather extremist approach stems from the realization that each user's needs, when sharing content on the Internet, are unique even when very similar to those of other users: if, in a material culture view, users make the Internet as much as the Internet (as part of other 'stuff') makes them (Miller 2009), the IndieWeb movement captures — within the domain of Read/Write Internet practices — material culture's 'commitment to keep in touch simultaneously with the extremes of universalism and particularism in modern life' (ibid., p9). It does so by acknowledging and interpreting users' desires to embrace the large-scale software environments — such as corporate SNSes — that allow them to be part of conversations and networks at a global scale, while enabling individual users to inflect in their own personal ways the technical and material details of how being part of these networks is performed, which often has little to do with the technical itself and is more closely connected to each individual's ways of relating to everyday objects around them, whether tangible and intangible, which all contribute to the material grounding of everyday life.

[82] The *Federated Social Web Summit* was an event held annually between 2010 and 2012, organized by the World Wide Web Consortium's (W3C) Federated Social Web Community Group; active developers of federated social web software such as those discussed in this chapter were invited to work together on ideas, protocols and implementations of federated platforms in order to foster shared aims and to promote interoperability between different platforms.

8.2.3 Balancing control and convenience: taking part in social conversations on the web

The choice of how to interface at a personal level with global-scale networks is explicitly left to individuals: as opposed to other groups'[83] calls to abandon hegemonic proprietary SNS silos and to replace them with open, federated SNSes, IndieWeb activists acknowledge that users may want to continue being part of conversations with relatives, friends and acquaintances who are only active on mainstream SNSes such as Facebook or Google+, while still retaining control over personal data.

The IndieWeb's suggested strategies to interface personal and social are centered around two main configurations, which can be combined to different degrees and used at the same time by each user according to their preferences, specific circumstances and technical skills.

The strategy more closely responding to the IndieWeb's tenet of 'owning one's own data' is labelled "POSSE" (Publish (on your) Own Site, Syndicate Elsewhere):

> It's a content publishing model that starts with posting content on your own domain first, then syndicating out copies to 3rd party services with [permalinks] back to the original on your site. [...] POSSE is about staying in touch with current friends now, rather than the potential of staying in touch with friends in the future. (IndieWebCamp 2015b)

Thanks to a multiplicity of plugins for the most widely used blog platforms and CMSes, content published by a user on their own website can be automatically and selectively published ('syndicated') to one or more mainstream SNSes; conversely, replies, mentions, upvotes ('likes') and comments may be posted back to the user's personal website and linked to the original post: the user's copy of their own posts and related conversations is the *canonical* copy, under full control of the user, whereas conversations happen wherever suits the other participants, who don't even need to know about the process that keeps the canonical copy 'in sync' with copies elsewhere.

A second, complementary strategy, is labelled "PESOS" (Publish Elsewhere, Syndicate (on your) Own Site) and revolves around the ability to collect and archive on a user's own website their personal content first published on social networks and elsewhere:

[83] Such as developers and users involved in federated social network projects.

> It's a Syndication Model where publishing flow starts with posting to 3rd party services, then using some infrastructure (e.g. feeds, pingbacks, webhooks) to create an archive copy under your domain. (IndieWebCamp 2015a)

Although these two strategies map to very different setups of the software code needed to interface with external services, as well as distinct uses of these services' APIs, the choice of how to use each (or a combination of the two, as is often the case) extends beyond the technical: as one of the key tenets of IndieWeb is to empower individual users to assemble their own web practices, its technical forms inscribe the ability to accommodate different kinds and levels of technical agency and of skills required from users, but most importantly personal preferences and intuition of acceptability as to how to relate to translation of social norms to interactions involving content exposed on the Internet.

Madianou and Miller (Madianou and Miller 2012, 2013) examine in close ethnographic detail how in recent years (essentially post-2010) users in an increasing range of contexts tend to base their choices of communication media over the Internet on 'the social and emotional consequences of choosing between those different media' (Madianou and Miller 2013, p170) rather than on technical constraints or monetary cost implications:

> As cost and access become less important, and as media literacies develop, then people start to see the reasons why any particular person has chosen any particular medium as a social act—something that in our studies is found to be fundamental in actually constituting that social relationship. (ibid., p183)

How IndieWeb users configure their strategies of interaction between their own personal website and external web services and apps similarly highlights that social norms and customs shape choices that from a technical stance would match equally well users' requirements.

This is specifically evident when considering the Read/Write character of the two IndieWeb strategies outlined above (POSSE and PESOS): the interaction between one's personal website and external services often channels two-directional social exchanges, by keeping synchronized across multiple (social) sites not only users' content but also the associated *conversations*, which are transparently relayed between, for example, Twitter (where most of an user's contacts may be posting) and the user's personal site (where the user may be following ongoing conversations, and posting her or his own updates):

> When I type a note into that form [on my own website] and hit "post", here's what happens:
>
> - I store the note in my own database.
> - I send a copy to Twitter as a status update.
> - Twitter returns [...] info about the tweet I just created.
> - I take the ID of that tweet and store it in my database along with the original note.
>
> Having the ID of the copy on Twitter allows me to provide some Twitter-specific actions from my own site: reply, retweet, fave, etc. (Keith 2014a)

From the list of steps described in detail in the user account above, it is also evident that strategies of Read/Write interaction with external services are inextricably dependent on the mediation of software code, which quietly records and updates in the background any links between the different sites where an user's content is stored, making it possible to later use these to update conversations across the network of sites involved: a multitude of atomic components of users' conversations and — ultimately — computational lifeworlds are incessantly synchronized and negotiated through algorithmic processes that *depend* on the ability to inteface with proprietary silos but are ultimately *configured* entirely by each IndieWeb user.

Whereas hand-crafting the specific code that allows one's website to implement and sustain POSSE and PESOS strategies is considered a sort of 'rite of passage' among IndieWeb users[84] and proudly described on many blog posts such as the one from which the excerpt above is taken, the centrality of this kind of multi-actor (human users/software code/APIs) conversations to most IndieWeb users' daily needs led to the creation of FLOSS web apps and services such as Bridgy[85], whose purpose is specifically to provide a reusable software formalization of the many handcrafted implementations created by IndieWeb users. Less technically inclined users can therefore simply configure a network of relationships between services they use within the Bridgy interface, which then transparently coordinates all the exchanges of data and content needed to keep al the user's contacts updated as conversations unfold, allowing the user to focus on the social aspect of their practices online rather than on technical details, but still trusting their own website rather than a proprietary platform as the canonical repository of personal and social content relevant to them.

Likewise, although the blogs of participants in IndieWebCamp events often describe how their IndieWeb strategies include a variety of heterogeneous software tools carefully selected and customized to work together in ways that address each user's unique needs, some developers recognized that such intimate connection with the software used is not always possible (depending on technical skills and other constraints) or desirable (some IndieWeb users may just be happy to focus on content and social interactions without having to handcraft their immediate software environment), leading to the development of several web services that allow less technically minded users to create their own 'home' on the Read/Write Internet while delegating content storage, connection to external services and other implementation details to software libraries whose functioning is not questioned and that don't need any manual assembling. Interestingly, one of the most widely used such systems at the time of writing is Known[86], whose lead author, Ben Werdmuller, is also the original creator of the Elgg social networking software that ended up being the only actual 'Web 2.0' part of my development project discussed in Chapter 5, plotting a (personal) development trajectory that cuts across Web 2.0 and the computational turn of the Internet, identifying in the motivations and strategies of IndieWeb a favourable context in which the lifeworlds of common users could meaningfully include Read/Write Internet affordances without the technical constraints that hindered similar efforts within the Web 2.0 context.

8.2.4 IndieWeb hackers: motivations and peer support

By explicitly portraying itself as a *movement* focused on identifying and promoting pragmatic and viable ways of addressing its core principles, rather than as a project primarily focused on developing technical solutions, the IndieWeb movement's primary role can

[84]This is also reflected in the importance given to the ability to 'federate IndieWeb conversations' within the *IndieWebify.me* guide (`http://indiewebify.me`), one of the popular initial information points for Internet users interested in 'getting on the IndieWeb', which groups under the label 'Federating IndieWeb Conversations' the practices (including integrating conversations across external services) that require more complex technical configurations that users are invited to explore after having added more basic IndieWeb functionality to their websites (Novak, Walters, et al. 2015).

[85]`https://www.brid.gy/`.

[86]`https://withknown.org/`.

be considered that of identifying mismatches between existing narratives (both the hegemonic corporate ones as well as the more technical-focused counter-narratives examined in the rest of this chapter) on one side and individual users' needs on the other, giving a simple and recognizable name to the array of technical solutions available to overcome these mismatches.

The IndieWeb movement is an atypical community of developers and users: membership is open and porous and motivated by knowledge (improving the shared understanding of alternatives to Internet practices not specifically focused on individual users, and promoting their uptake) rather than by roles (developers, designers, etc.) and there is limited interest in developing canonical tools; rather, existing tools proposed by members and new ones being developed are assessed according to their ability to respond to the movement's core principles, while the choice of actual tools to use is left to individuals. Accordingly, the IndieWeb movement cannot be considered strictly either a development project (there is no defined set of software projects being collectively developed, although individual members develop and use their own IndieWeb projects or occasionally collaborate on shared projects) nor an organization focused on creating standards, even if informals (over time, best practices and a few key protocols related to authentication and personal data exchange have been identified and developed, but choice of practices and strategies is ultimately left to each IndieWeb user): it may be conceived more accurately as a *community of practice* (Wenger 1999), loosely connected over common goals and principles:

> With IndieWebCamp we've specifically chosen to encourage and embrace a diversity of approaches & implementations. This background makes the IndieWeb stronger and more resilient than any one (often monoculture) approach. One of the key things we recognize with IndieWebCamp is that no one project is likely to be *the* answer. We're much more likely to advance the state of the art by encouraging everyone to build what works for them, and then figure out how to interoperate between different coding/implementation approaches. This is what makes IndieWebCamp different (more inclusive) than all other such "open source" efforts out there. (IndieWebCamp 2014b)

As a consequence of its explicit focus on the individual (user, developer, website 'owner') and of the heterogeneous character and ephemeral p2p configuration of the social and technical interactions linked to it, the IndieWeb phenomenon is inherently difficult to oversee. As no central coordination is required for interactions to happen between each IndieWeb site and its linked external silos (SNSes, etc.) — as well as with other IndieWeb sites, records of network activity between its nodes are only available to the nodes themselves, making it impossible to gather reliable data about the extent of the IndieWeb 'network' as a whole. Besides its core human group, visible through a small but lively IRC chat room and at occasional in-person gatherings, what can be observed is the numeric extent of the publicly available software artifacts and services used as part of individual IndieWeb strategies; this is substantially similar to the non-overseeability of p2p filesharing phenomena, although in the case of IndieWeb no equivalent of BitTorrent's trackers and distributed databases (DHT) of content available across nodes can be used to infer estimates of size and shape of the network. Beyond the research implications (gathering data on IndieWeb users is essentially a manual task, only partly open to 'snowballing' methods), this is manifest in the absence of any substantial sociality and socialization structure of *computational nature* (such as those informing SNSes) on the IndieWeb: discovering 'peers' is largely a preoccupation external to the computational structures of IndieWeb. Some limited initial attempts at building external 'social discovery' processes have been made by IndieWeb users: the *indie-stats* project[87] is a simple web crawler that fetches web pages and analyzes the HTML source code looking for markers of IndieWeb metadata, compiling a list of websites found to implement some form of IndieWeb strategies; it is, however, mainly aimed at mapping technical strategies in use rather than at providing an interface to allow discovery of social contacts: initial design for this project came from a blog post of the lead developer of the Bridgy software discussed above, who lamented the scarcity of data on actual IndieWeb technologies in use:

> When I have to decide whether to implement a feature in Bridgy, or how to prioritize tasks, I often make assumptions like *most indie web sites have an h-card, or PSCs and PSLs never got much traction*. I know they're based on anecdotal evidence, not actual data, but it's all I have, so I run with it. Clearly not ideal. I'd love to use real data instead! Here's a project idea: crawl indieweb sites and generate usage stats for microformats2 classes and other indieweb features (Barrett 2014).

In absence of centralized and algorithmic social *networking* (boyd and Ellison 2007,

[87] `https://github.com/bear/indie-stats/`

p211) tools, *general* socialization strategies (as opposed to establishing connections with other individuals interested in IndieWeb) are therefore left to individual users, and range from exploiting the social graphs of the mainstream SNSes (Twitter, Facebook, etc.) to which a user's IndieWeb site cross-posts updates, to non-digital mutual introductions. The key point regarding socialization, though, is that while certainly not inherently *anti*-social, the IndieWeb counter-narrative is focused firstly on the empowerment of individuals to publish personal content online and is essentially agnostic, by design, on how individuals get to know other users with whom they may want to interact online.

Whereas external social graphs are routinely used for general socialization, socialization opportunities *within* the IndieWeb movement itself (getting to know other IndieWeb site owners) rely mainly on in-person meetings and conventions; besides the annual IndieWebCamp gatherings discussed above, of central importance are the fortnightly Homebrew Website Club informal gatherings, coordinated across several worldwide cities through the central IndieWebCamp wiki. The aim of these meetings is to provide an opportunity for direct contact between people interested in the IndieWeb, through which expertise, advice and mutual help can be exchanged. The name clearly references the Homebrew Computer Club gatherings initiated in the Bay Area in 1975, and these are in fact explicitly acknowledged as the initial inspiration, with similarities extending to practical matters such as timing, frequency and structure of the gatherings with the aim of making these as much as possible part of the stream of daily life rather than letting them being felt as one more technical gathering a person may want to attend (Çelik 2013). More importantly, the public accounts of participants to these meetings relate a close similarity to the passionate and informal experimentation spirit of the Homebrew Computer Club events, with a focus on building working prototypes, on helping each other overcome practical difficulties experienced while assembling personal IndieWeb strategies, and on developing a shared understanding of the values and distinctive traits of IndieWeb through show-and-tell sessions:

> But most of all, this indie web sort of movement, this indie web group, is kind of like just a support group; it's like a bunch of people helping each other out, and it's really good fun, and we get together in real life and we hack on stuff, we discuss stuff, we help each other out, and for me it's com-

> pletely selfish; I'm not setting out to change the world. I should point that out. This is not a mainstream movement; this is very much a niche thing right now, and I'm OK with that, because all I care about frankly is my own website. (Keith 2014b)

The IndieWeb's focus on strategies relevant to individuals while fostering a sense of community and peer help has also started creating spaces for the exploration of business models alternative to the exchange of privacy for convenience for which major SNSes are often criticised: besides the hosted services mentioned above (e.g. Known), which are structured like most small independent Internet service providers, but focused on IndieWeb services and on a public interested specifically in online publishing through these, nascent enterprises such as IndieHosters[88] and other similar ones provide *personal support*—rather than self–service hosting—as their core business. Whereas the aim is similar to that of other hosted services, each new client is invited to choose an *host* (which in this case is a *person* — one of the business partners — rather than an *internet server host*) who becomes their first point of contact for any help they may need while developing their IndieWeb strategies and personal infrastructure: technical expertise and strategic guidance are the services provided, rather than just infrastructure. Although the viability of such businesses needs to be assessed on a longer timescale, this alternative to the rent-focused 'monetization' attempts of mainstream Internet businesses and startups recognizes the central value of human expertise and of personal contact as empowering factors for less technically minded users who are interested in developing their own Read/Write Internet strategies while lacking technical expertise; interestingly, some of the students interviewed in my fieldwork explicitly stated that in their attempts to curate their personal content online without accepting the privacy tradeoffs of major SNSes, what they valued most was not the availability of suitable technical infrastructure (which was substantially taken for granted) but the ability to rely on the support of family members, friends or acquaintances with specific technical skills.

While trying to assess the adoption of IndieWeb practices, it should be noted that self-hosted blogs — popular amongst free software and web developers — even without any of the specific metadata markers used by IndieWeb sites, fall within the digital self-reliance

[88] `https://indiehosters.net/`.

spirit of IndieWeb although their owners may not be interested in identifying them as such, nor in identifying themselves as part of the IndieWeb movement, further blurring boundaries and extent of the phenomenon, and at the same time highlighting both the relative simplicity of its basic arrangements even within the overall dominance of centralized infrastructures for publishing of personal content and information.

Similarly, several of the student stories discussed in Chapter 6 outline a commitment to 'indie' formats and strategies for the publishing of personal content online, with motivations ranging from interest in close control over aesthetic form, to concerns regarding privacy, to convenience. This may seem counter-intuitive (as mainstream social networks are continuously optimised to foster 'frictionless sharing'), but in the accounts of some the students interviewed the *social* — rather than technical — aspects of what constitutes convenience in everyday practices emerged: this seems to correspond with the trends analyzed by Madianou and Miller (Madianou and Miller 2012, 2013) and outline a real interest for content publishing and sharing strategies modelled closely after users' preferences and everyday practices, even when mainstream SNSes offer a technically more convenient user experience.

8.3 Decentralized Read/Write Internet: reassembling p2p infrastructure

> [W]e need a Web that is reliable, a Web that is private, while keeping the Web fun. I believe it is time to take that next step: I believe we can now build a Web reliable, private and fun all at the same time. To get these features, we need to build a 'Distributed Web'. (Kahle 2015)

If the actors involved in personal accounts, media coverage and scholarly analysis of uses of the early (pre-2000) Internet in everyday life were mainly the users themselves, the websites or Internet systems (IRC, virtual worlds, etc.) on which their practices took place, and *genres* of Internet activity, as the Web 2.0 narrative unfolded throughout the 2000s decade a new class of actors became increasingly central to these same accounts: *platforms* (Jenkins, Ford, et al. 2013) such as Facebook, Google, Twitter, or '*stacks*' (Madrigal 2012; Sterling 2013) such as Google, Apple and Microsoft — and, in different ways and less visibly for end users than for Internet developers, Amazon. Whereas these still manifest themselves to users as websites and web apps—or a combination of websites/apps and mobile apps and services in the case of stacks—their role within a vast proportion of contemporary Internet practices goes well beyond that of content gateways that Web 2.0 websites and apps had: their centrality to daily Internet practices stems from their ability to provide services whose underlying computational complexity makes the extent of functionality provided to users essentially non-fungible, unless through infrastructures similarly complex to those of major Internet corporations.

The ability of these companies to provide convenient Internet services coordinated through multiple devices that may belong to a same user, thanks to their deep and broad computational capacity, constitutes a new and qualitatively different type of centralization of configurations of Internet infrastructure (and, most importantly, of data flows and of user practices) than what I outlined in Chapter 4 while discussing client/server and centralized configurations built on top of the essentially p2p infrastructure of the Internet: the centralization power of these corporations is also consolidated through their control of several layers (hence the label 'stacks') of the computational infrastructure that provides convenient services to end users: as an example, Google manages—for its internal use, for developers and for end users—cloud infrastructure, big data services, advertising services, analytics, social graphs (the Google Plus social network and the social graphs behind its other user services such as YouTube), content and partnerships with media corporations (YouTube, etc.), mobile and desktop operating systems and app stores (Android, Chrome OS), etc. The computational complexity that would be required across such a broad spectrum of layers of Internet infrastructure in order to provide similar integrated user experiences as those offered by any of these oligopolistic corporations is not attainable by smaller or independent organizations: the gatekeeper role of these corporations is therefore increasingly more problematic, leading to extensive questioning of their power by Internet activists, journalists and concerned individuals and organizations.

8.3.1 Countering centralization: hacker motivations

In response to such concerns, a multitude of projects have been started in recent years by hackers and scholars who aim to 'redecentralize' the many traits of Internet applications and practices for which the existence of central coordinating infrastructure and servers is not an unavoidable configuration. Hackers' motivations for such endeavours include pressing concerns about privacy of personal data, interest in avoiding the implicit exploitation of user labour and data for commercial gain, avoidance of state and corporate surveillance, and interest in democratization of Read/Write Internet infrastructure by allowing users to exercise choice and to contribute ideas for development.

Whereas scholarly research has been developing an extensive corpus of critique to the power struggles, labour and surveillance issues linked to Web 2.0, user-generated content and social network sites (Lovink 2011; Petersen 2008; Sandoval 2012; Terranova 2004; Van Dijck and Nieborg 2009, amongst many other contributions), a common trope in hacker circles summarising these issues as manifest in major Internet platforms is often expressed through a short quote commonly attributed to user 'blue_beetle' on metafilter.com: 'If you are not paying for it, you're not the customer; you're the product being sold.' (blue_beetle 2010). Hackers involved in the projects surveyed for this chapter further elaborate this trope in their projects' manifestos or in public comments and remarks; Aral Balkan, co-founder of the ind.ie p2p messaging platform and one of the hackers most vocally concerned about the exploitation of Internet users inscribed in the political economy of 'Silicon Valley startups', remarks, echoing Zuboff's (2015) stance (which he explicitly cites elsewhere in his writings):

> We are the products that these companies... these corporations, sell to their actual customers. Now, in the past, and unfortunately in some places still today, we had the practice of selling people and we called this ”slavery”. It was the act of selling people's bodies. I think we've reached a point where we have to ask ourselves the very uncomfortable question of: what do we call the business of selling everything else about you that makes you who you are apart from your body? Because this, for the most part, is the business model of mainstream technology today. It's the business model of Silicon Valley today. (Balkan 2015)

Balkan's motivation throughout his work on decentralized personal communication infrastructure discloses a multifaceted concern shared with several other hackers involved in redecentralization projects: firstly, the focus is specifically on technologies that relate to the intimate, everyday life sphere of individuals ('everything else about you that makes you who you are apart from your body'); secondly, the rendition of control over intimacy and privacy is problematized, and considered through the process of trading this control in exchange for convenience and for the ability to afford the use of technologies offered as neutrally 'at-hand'; thirdly, this illusory neutrality of the technologies offered is contrasted with the economic imperative for startups — through business models focused on short-term returns and 'founder exits'[89] — to maximise short-term valuations by locking users into convenient centralized platforms.

Developing decentralized alternatives can as well constitute an attractive intellectual challenge by itself — an important motivating factor for hackers (Coleman 2012a, chapter 3) — through the exploration, implementation and improvement of configurations that could offer similar convenience, robustness and performance as their mainstream centralized counterparts, while avoiding the extreme knowledge and power asymmetries imposed by the use of major Internet platforms.

Besides the general interest and relevance of decentralized alternatives to incumbent platforms, these alternatives are specifically relevant in the context of this dissertation in two ways: firstly, decentralized practices were explicitly discussed and embraced by some of the more technically competent students interviewed for my fieldwork, while other students discussed concerns or frustrations that could partly be limited by the use of alternatives to the mainstream applications they reported using; almost every student, moreover, disclosed using at least some widespread decentralized configurations such as p2p file sharing and discussed the impact on their daily practices of the ability to avoid cen-

[89]'Exit' in tech startup circles is the process through which startup founders sell their shares/equity in their own company to an unrelated party; this is typically an investor (or group of investors) who are interested in the fast growth of successful startups as part of their investment strategies. Despite the fact that exits may not constitute a very favourable financial deal in the longer term for startup founders—as a fast–growing startup may generate in a short timespan higher profits than the amount of cash paid by the buyers at the time of founder exit—this strategy is nevertheless considered highly desirable by several founders, who see themselves as better skilled at *starting* fast–growing companies than at managing them in the longer term, therefore seeking to reap short–term financial benefits and trying to finance successive startups through the earnings of previous exits. Such strategies are considered by many of the hackers surveyed in this chapter particularly unethical as they involve acquiring customers through the promise of delivering a valuable and technically excellent service, whereas once control over the company is relinquished in exchange for cash, the new investors are typically driven mainly by the desire to maximise profits, often by exploting user attention and loyalty for widely different purposes than the original founders.

tral coordination/control. Secondly, although still far from any substantial mainstream adoption, some of the alternatives discussed in this section engage with issues of user convenience and computational complexity that the project discussed in Chapter 5 struggled with, while attempting to build early alternatives to the rising centralization inscribed in initial Web 2.0 applications: these more recent alternative rationalizations based around decentralized configurations are here analyzed within the frame of the computational turn of the Internet, assessing how the vastly different context in which they are developed affects their ability to overcome the failures of those earlier attempts.

8.3.2 Computational turn and decentralization: from user-to-user to machine-to-machine configurations

It is important to outline here how decentralized Internet infrastructure and practices differ from (and partly overlap with) those of IndieWeb discussed in the previous section. Whereas IndieWeb users are mainly concerned — at least for the portions of their Internet practices involving sharing of personal content — with retaining control over their own *content*, while generally accepting to delegate the mechanics of *conversations* about content to incumbent SNSes according to the strategies outlined earlier (POSSE/PESOS), decentralization strategies are instead actively concerned with conversations and exchange of data, and the ways in which network topology influences the ability to carry on everyday conversations and tasks over the Internet by exchanging data directly with the devices of correspondents.

This focus on conversations and exchange of data can be further deconstructed into two main contexts, which help differentiate recent decentralization efforts from earlier such strategies. Central concerns of pre-computational turn decentralized projects were the ability for individual users to publish and exchange content without having to rely on the coordination of proprietary services and, if using services operated by third parties because of convenience, to be able to choose between essentially fungible providers and to switch between them with little or no disruption[90] , if needed: in brief, the context is that of *user-to-user* communication. As users' Internet practices increasingly span over multiple personal devices, however, exchanges of data that need to be accounted for are not anymore mainly user-to-user but also *machine-to-machine*, whereby a single user's data, content and conversation fragments are often generated and consulted on different devices (e.g. photos and videos taken on a smartphone, edited on a laptop, shown to a family member on their own tablet, etc.), requiring constant, reliable and unattended synchronization, as well as consistent management of access to resources (typically using a single user account across all personal devices, and to access different services on each of them). Mainstream solutions, developed alongside the increasing diffusion of new kinds of second or third personal digital devices, have so far evolved around single-vendor 'stacks', which span on-device services (e.g. synchronization apps on mobile devices), infrastructure (e.g. proprietary synchronization APIs and storage distributed across vendor datacentres) and web user interfaces or mobile apps. A further, hybrid context can be identified in small groups, such as a household or a group of close friends and relatives, where everyday life content (documents, photos, emails, secrets, videos, etc.) is shared between closely connected individuals, who may be using a combination of personal (digital) accounts and services and shared ones, typically within a single space (a family home) or across a few personal spaces. Examples of Internet practices involving such small groups are particularly interesting and relevant for my research as they emerged during my conversations with fieldwork participants (for example, students wishing to privately share messages and photos with family members abroad or far from London); moreover, they constitute a typical context in which the physical and relational proximity of individuals involved makes the need to use centralized infrastructure to share content visibly as problematic as it is in the context of synchronization between devices belonging to a single individual through machine-to-machine exchanges. Although far from new (as many examples are discussed in key ethnographies of computer and Internet users in the home environment: Bakardjieva 2005; Lally 2002; Miller and Slater 2000, etc.), these contexts acquire a distinct new character with the computationalization of Read/Write Internet, which both facilitated the rise of major structures of central coordination, while on the other hand making available to a wider public tools, strategies and knowledge required to reassemble decentralized alternatives to the incumbent centralized organizations.

[90]As an early decentralized core infrastructure based on a mix of client-server and federated configurations, an

Finally, it is also useful to articulate a further distinction: the strategies and configurations discussed in the present section are specifically analyzed in their role as enablers of conversations between individuals and exchanges of data and content *within the private sphere*, according to the focus of my research and to the use contexts discussed by my research participants. Several — if not all — of these strategies and configurations are *also* useful in (or have even been created within) the more widely explored contexts of critical/tactical/alternative media/journalism and online activism (Allan 2006; Coleman 2014; McDonald 2015; Milan 2013; Olson 2012) that make use of decentralized Internet communication tools to try to circumvent surveillance and censorship; they are here analyzed, nevertheless, mainly in their role within the sphere of personal communications.

8.3.3 Redecentralized Internet: strategies and configurations

Whereas the overarching strategy of redecentralization efforts can be considered substantially similar across different projects — the avoidance of mainstream client/server architectures relegating power and knowledge to the entities operating the server part of these architectures — in practice each different context presents distinctive conditions and strategies that in turn are inscribed in specific Read/Write affordances and are supported by specific computational configurations. These different contexts are briefly outlined in the following paragraphs, before moving on to analyzing common challenges and implications on power imbalances and agency of developers and end users in the final section of the chapter.

8.3.3.1 p2p messaging

Direct exchange of text, audio and video messages, whether asynchronously (for example, email or Internet instant messages used alternatively to SMS text messages) or in real time (audio/video chat) is one of the simplest forms of personal communication whose topology usually involves small groups of mutually known users[91]; whereas a direct link

illustrative example is that of sending and receiving email through a desktop email client (i.e. not through webmail cloud services): most home users would typically rely on email server infrastructure provided by their ISP, but as the underlying protocols (SMTP, IMAP) allow standards-based interoperability across any email infrastructure, users can easily switch to an alternative provider of email services.

between the parties involved in these conversations may seem natural given their direct mutual acquaintance, mainstream configurations enabling these forms of communication are commonly based on client/server or federated[92] topologies.

Recent alternatives to client/server architectures aim at redressing the power imbalances inscribed in these centralized mainstream configurations in order to open spaces for private Read/Write strategies within this domain of lifeworld Internet; these alternatives rely on one of the following two decentralized configurations:

1. *p2p, direct connection between the devices of participants in conversations.*

An established project based on this strategy is the p2p communications suite RetroShare[93], which has been developed as a free software project since 2006, aiming to build a secure and private 'F2F' (friend-to-friend, in the project developers' own words) communications infrastructure lacking any centralized coordination and oversight:

> Retroshare tells you about the people around you: your friends and–optionally–friends of your friends, but very little about what lies beyond. You can receive information like forum posts and files from the rest of the network, but you have no idea about the original source of the information. Retroshare's design ensures you have little idea of who is out there – it is just friends, of friends, of friends, of friends ad infinitum. (from the 'Ideals behind RetroShare' blog post: drbob07 2012)

A peculiarity of RetroShare's configuration is that it relies on a social graph (boyd and Ellison 2007), similarly to SNSes, to make content and conversations meaningfully accessible to users, effectively combining direct p2p connections as the opaque, infrastructural machine-to-machine configuration, with a more familiar social network representation of connections as visible interface for human users.

A more recent pure p2p project is the Tox encrypted text/audio/video messaging protocol[94], which is part of the 'post-Snowden' set of projects aimed at enabling avoidance of central control and surveillance for personal communications, as also hinted at through the project's enthusiastic motto ('Working for a more secure world'). Although features of applications based on the Tox protocol are similar to those of RetroShare, the architecture

[91]This can be contrasted, for example, with contexts involving potentially large numbers of users, mostly unknown to each other at least in person: forums, discussion groups, commented livestreams of videogame play where a 'celebrity' player broadcasts to a large audience, virtual worlds and large multiplayer games, etc.

[92]See below for a brief description of federated infrastructures.

[93]`http://retroshare.sourceforge.net/`.

of the Tox system more clearly articulates a p2p data exchange layer (formalized as the Tox protocol itself) and a user interface layer, which consists of independently developed applications, interoperable as far as they rely on the underlying Tox protocol. Given its close focus on privacy and safety of users, social graph functions are absent from the core protocol and delegated to the user-facing apps or to external systems, such as directories to which users can sign up voluntarily.

2. *Independent federated configurations.*

Federated infrastructures typically consist of a combination of client/server connections between end users and their reference server, and direct p2p connections between servers. Email exchanges rely on a federated infrastructure, although one of the distinguishing traits of the projects discussed in this section is that they actively encourage users to set up their own server (to which their devices connect, and that in turn connects to other users' federated services) rather than relying on servers belonging to corporate entities. Most of the recent decentralized projects based on this configuration rely on a single underlying data exchange protocol (XMPP), which has been used since 1999 as the infrastructural foundation of the Jabber instant messaging network, as well as for other applications such as machine-to-machine data exchange in Internet of Things projects. On one hand, the use of a standard protocol allows users of different applications to communicate with each other, and this has in turn enabled XMPP-based applications to reach wide adoption, despite not being part of large-scale mainstream services; instant messaging systems of major Internet corporations (Facebook, Google, Whatsapp, amongst others) rely on this protocol, and Google contributed a standard extension allowing voice and video conversations to be carried over the XMPP protocol (which had been allowing only text conversations until then), although most of the services offered by mainstream corporations remove the ability for their (otherwise standards-compliant) services to communicate with users of external XMPP-based services, artificially removing the ability to join part of a larger federated network. On the other hand, however, whilst a variety of free software XMPP servers are available (and sometimes even included in 'turn-key' personal internet services), installing, configuring and maintaining an independent XMPP server

[94] `https://tox.chat/`.

often requires dedicated systems administration skills: accordingly, running a fully independent communications infrastructure through many of the decentralized softwares using this specific federated configuration is not a realistic prospect as part of personal communication strategies for common Internet users. Moreover, although the underlying data exchange protocol allows interoperability between different implementations, in practice valuable protocol extensions (such as support for end-to-end encrypted communications as available in the CryptoCat[95] project) require that all users wishing to engage in a conversation whilst relying on specific features use the same software on their devices.

8.3.3.2 Storage and direct sharing of personal content

Besides interpersonal communications, lifeworld Internet technologies include — often at a very intimate level — storage, management and sharing of personal content; similarly to small-groups communications topologies, care for personal content can intuitively be associated to a direct connection between the devices storing personal data, yet most of the mainstream configurations depend on central coordination and 'cloud storage' provided by private companies behind opaque APIs and often with little transparency about which level of access employees (and therefore, potentially, government agencies) have to content stored on the company's servers on behalf of users[96].

Moreover, personal content may not even be permanently stored by users in any immediately retrievable and reusable form on their own devices: part of the early Web 2.0 enthusiasm about production and sharing of content through specialized web user interfaces effectively relied on the choice of the providers of Web 2.0 apps to store the 'canonical copy' of users' content internally (rather than storing this on users' own devices), on storage infrastructure closely connected, for performance and strategic reasons[97], to the web applications accessed by users. Even when exporting content is possible, this is not

[95] `https://crypto.cat/`.

[96] To cite an example of such concerns that was subsequently publicized widely in mainstream media, spurious encryption practices by cloud hosting provider Dropbox were exposed by security researcher Soghoian in a blog post (Soghoian 2011) analyzing how algorithms chosen by Dropbox to minimize datacentre storage costs could only rely on the ability to manage user content in its unencrypted state even when the actual storage was encrypted (although using a single encryption key to which Dropbox had to have access, rather than using per-account encryption keys only known to end users).

usually a daily concern for users: the extent of incidents through which personal content has been lost because of web applications ceasing operations or losing user data has been mentioned earlier in this chapter while referencing the informal backup activities of the volunteer Archive Team collective.

Accordingly, decentralized alternatives to mainstream content storage seek to develop Read/Write configurations within this domain of lifeworld Internet, with the aim of addressing the two core issues a) of direct control over the canonical location of content (similarly to IndieWeb projects discussed earlier) and b) of direct sharing of content with the intended recipients.

Whereas storage of personal content is not inherently a concern related to Internet practices, it becomes so inasmuch as users rely on Internet applications to share it with others and to synchronize it between personal devices. An overview of the two main variants of decentralized Internet applications focused on letting users store and manage the canonical copy of their own content ostensibly highlights the technical and cultural barriers that are often cited as a compelling reason why users routinely choose centralized alternatives despite their associated privacy and data safety concerns, and also shows a conflicting interdependence between the two aims of retaining control over content and being able to easily share it with others: on one hand, web applications such as MediaGoblin[98] have developed functional and convenient ways to host personal content and media (text, videos, photos, illustrations, 3D models, etc.) and to selectively share each item or group of items through the web interface of the application, as part of their key aims, as described by MediaGoblin's lead developer:

> [I]f you think about media publishing as a gallery, I mean, I started this because I'm a hybrid programmer and artist and I wanted a place where I could put a bunch of my own works, myself, all in one place. So you know, the closest thing that comes to doing this sort of thing that has multiple types of media is — DeviantArt does this in some ways, Flickr kind of does this [...] But I mean, the reasons why somebody might want such a thing is [...] is trying to give power back to people and trying to get — to take things out of these gigantic mega-sites. (Webber, in Bolychevsky 2013b)

[97] For example, to allow the operator of a Web 2.0 application to infer metadata from the combined set of user content managed by the app, in order to provide services to users, such as suggestions for browsing of potentially interesting content by other users, but also for commercial exploitation of user metadata through targeted advertising, etc.

Installing and managing these web applications, however, requires specific technical expertise; although users could instead rely on a media server managed by a trusted person, this reintroduces a degree of indirection over control on personal content. On the other hand, software programs such as Camlistore[99] (whose ambitious motto is, eloquently, 'your personal storage system for life') or Syncthing and git-annex[100] are designed to be run primarily on a user's own personal devices at home (rather than on a server/virtual machine in a data centre), keeping content closer to users' ability to manipulate it directly (as files on a local computer, rather than through an opaque web interface on top of remote storage), although selectively and meaningfully sharing individual items with others, especially outside of the local home network, is a cumbersome process that requires that users set up personal ad hoc strategies outside of the software used to manage data locally.

Whereas some of the p2p communication strategies discussed in the previous section are routinely used by common users with little practical difficulties, p2p content storage and sharing strategies are at the moment still the domain of technically knowledgeable, determined early adopters. One notable exception is (semi)anonymous p2p filesharing through established protocols and apps such as BitTorrent, which most of the students I interviewed for my fieldwork admitted to using (see Chapter 6); although some of the applications discussed above (notably, Syncthing) employ strategies very similar to those of BitTorrent to help users connect to other users and share content with them, the processes involved are still not as widely familiar to common users, thereby limiting the ability of most users to adopt decentralized strategies within this domain of lifeworld Internet.

[98] `http://mediagoblin.org/`.

[99] `http://camlistore.org/`.

[100] `https://syncthing.net/` and `http://git-annex.branchable.com/`: these are p2p alternatives to cloud-based data synchronization services relying on centralized infrastructure such as DropBox or Apple/Microsoft/Google's synchronization services.

8.4 Alternative rationalizations and computational mediation

Having outlined the essential traits of the main alternative rationalizations to mainstream Read/Write Internet configurations in the previous two sections of this chapter, I will now turn to discussing challenges shared amongst redecentralization attempts; in this final section, I argue that by developing alternative rationalizations within the constraints imposed by the legacy of centralized computational configurations inscribed both in mainstream Internet software engineering and in representations on which common users rely, hackers act as computational mediators, translating concepts and representations between end users, fellow hackers and technical infrastructure.

8.4.1 Redecentralization efforts and end user engagement

A common challenge discussed by developers of redecentralization projects is that of user engagement — the ability for such projects to attain any significant uptake amongst common users. On one hand, hackers are increasingly aware and vocal both about the issues they perceive within mainstream Read/Write Internet and about alternative rationalizations as partial responses to these issues; I could observe this throughout the years of exposure to hacker discourses and related analysis of texts, and this is also increasingly visible through the proliferation of hand-compiled directories of decentralized projects[101]. Common users, however — except those more firmly committed to retaining some degree of control and privacy over their Internet practices — are less likely to be aware of alternatives, doubtful about their viability, or resigned about the tradeoff between privacy and convenience implicit in the use of apps, services and devices provided by major Internet companies (Rule 2007; Turow, Hennessy, et al. 2015).

[101] To cite the most visible examples at the time of writing: the "PRISM Break" website (`https://prism-break.org/`) was started in response to the surveillance programmes exposed by Snowden and is a comprehensive and constantly updated directory of Internet services and software aimed at helping users to "[o]pt out of global data surveillance programs like PRISM, XKeyscore and Tempora"; the Redecentralize project (`http://redecentralize.org/`) maintains a similar directory (`https://github.com/redecentralize/alternative-internet`), more oriented towards infrastructural projects ("A collection of interesting new networks and tech aiming at decentralisation (in some form)") rather than personal apps, although both lists span over personal and infrastructural projects.

Hull (2015) analyzes the power issues restraining user agency on Internet privacy, highlighting how the leading privacy model on the Internet ('privacy self–management') deceptively frames privacy management as a space for choices left to the individual, without exposing the wider implications of user choices nor effective alternatives beyond the range of settings provided to users or the option to avoid using an Internet service altogether. Similarly, the convergence of a) increasing computational complexity involved in even the apparently simplest daily tasks and b) increasing opacity of this same computational complexity (Berry 2014, ch.3), hidden beneath simplified user interfaces, mobile apps and unadorned mobile-first websites, is contributing to situating the understanding of the technical rationality of everyday Internet use further away from common users. Accordingly, hackers involved in alternative rationalizations show an increasingly pressing awareness of the difficult task of successfully promoting alternative framings to the hegemonic ones: user convenience, usability of alternatives and — most importantly — the ability to provide understandable representations of the issues these alternatives aim at addressing are common concerns highlighted.

The extent of the struggles faced by alternative rationalizations while trying to become viable alternatives for common users is evidenced by the data available for one of the decentralized SNS software most widely discussed in mainstream media beyond specialist forums, the diaspora* project[102]. Started by four undergraduate students at NYU in 2010 with the aim of building a federated, privacy-aware alternative to leading SNSes, the project attracted widespread interest on the Kickstarter crowdfunding platform, eventually collecting over 200,000 USD in funding while having set a target of only 10,000 USD[103]; as one of the few such decentralized projects for which partial (voluntary) usage data is publicly available[104], the total number of active users[105] of the surveyed part of the network in December 2015 was just less than 50,000: although many more may be active on unsurveyed portions of the network (for example, no data is available for the first server initially accessible only to crowdfunding backers), this figure can give at least a very partial sense of the vast difference of scale compared to major social networks[106] and, consequently, of the scale of challenges faced by alternative rationalizations.

Despite the extent of challenges, hackers working on decentralized projects often show a remarkable dedication to making their software usable for a wider public, however incremental and evolutionary these advances may be. Even though the economic forces involved are so widely disproportionate as to rationally suggest that alternatives hold close to no hope of becoming relevant, the complexity of the social-technical networks involved in computational Read/Write Internet, including the domains of governance and of education/learning, as well as the recursive public of hackers outlined in the previous chapter, make capital an undeniably fundamental but not unique terrain of struggle, as Feenberg remarks:

> [a]n exclusive emphasis on political economy tends to overestimate the rationality and coherence of capitalist strategies and to underestimate the significance of resistances, innovations and reforms in every domain except class struggle, where, unfortunately, there is little to report. (Feenberg 2002, p23)

As the work and consciousness of redecentralization hackers progressively extend beyond the self-referential concerns of a recursive public to reach out to common users interested in liberating their daily practices connected to the Internet, attention towards 'outreach' aims traditionally subordinate to the centrality of technical soundness of design, code and cryptography/security implementations is increasingly visible in hacker discourses. A concern that becomes relevant as alternatives start addressing the use cases of common users is that of ensuring that these are able to autonomously install and manage software:

> [ArkOS is a project] to put a focus back on creating tools that allow people to use the Internet in a decentralized manner [...] And it's not just... it's only half the battle, to produce *tools* and do these things. a huge, very important piece is making sure that people are able to use these things, that are not [...] system administrators and Linux... that have, you know, have been spending ten years in this field [...] it really needs to be made usable on an interface

[102] `https://diasporafoundation.org/`.

[103] Salzberg 2010.

[104] This data can be consulted via the Pod Uptime website (`https://podupti.me/`), which exposes a directory of servers ('pods') part of the diaspora* federated network, including number of users for each pod.

[105] Number of users active within the last six months on each pod.

[106] For comparison, Facebook reported 1.55 *billion* monthly active users as of 30 September 2015 (Facebook 2015).

> perspective and on an educational perspective for anyone that wants to use it. (Bolychevsky 2013a)

This was highlighted by several of my fieldwork participants as well: those who reported running personal services on their own virtual machines or who expressed interest in doing so mentioned how they often had to rely on ’technically knowledgeable’ friends or family members even when they had some significant experience with running such services themselves. A second emerging concern, often dismissed in the past in the context of recurrent critiques to FLOSS projects for lack of attention to (visual) design, is a wider attention to aesthetics as multifaceted visible/manipulable interface to the underlying computational complexity of technical implementations:

> [T]he products have to be beautiful; and by beautiful, I don’t mean aesthetics, I mean design. Holistically beautiful. We need to start thinking about design holistically in what we do, because we compete on experience in the consumer space, and if our products don’t, we will not be able to compete. [...] We need to wean people off of the current networks and the current social networks that they’re part of. We can’t just cut them off; we need to be social, and we need to be accessible. We need to reach a mainstream audience, and these all go together. (Balkan 2014)

Despite the ambiguous lexical choices of this specific quote, the intent — echoed throughout numerous statements and writings of other hackers, and visible through the work of Balkan himself and of his collaborators on the ind.ie project — is that of exploring opportunities within the domains of user interface, user experience (UX) and conceptual framing, in order to make the tangible affordances of decentralized projects understandable and usable for users increasingly accustomed to visual languages of Internet and mobile apps finely optimized through extensive research (for example, Google’s ‘material design’ visual language).

Although I have encountered, in developer discourses, only sparse evidence of self-reflective awareness of the transformation of their role through the focus on cognitive representations and meta-technical issues, the practices that emerge from the analysis of redecentralization projects can be interpreted as epiphenomena of a transition from a self-focused recursive public to a new kind of global public whose role is akin to that of *computational mediators*. In this capacity, redecentralization hackers can be seen as contributing to the *technical* reassembling of Internet infrastructure and user-facing affordances through configurations that try to address some of the power imbalances outlined in this chapter and emerging from the accounts of students analyzed in Chapter 6, as well as operating as *cultural* mediators (Bourdieu 1984) who perform two kinds of translations: they translate computational complexity into cognitive frameworks accessible to common users, and — inversely — they also translate users' struggles into a developing understanding of alternative rationalizations which are then reproduced through the spaces of hacker discourse outlined in the first part of this chapter.

8.4.2 Convenience and trust in decentralized configurations

As discussed in Chapter 4, centralized Internet configurations and client/server topologies have emerged over the years on top of the Internet's lowest level p2p/mesh infrastructure for a number of reasons; considering only the technical ones, for the moment, these can be broadly reconduced to the need to optimize how the many layers of infrastructure involved in the operations of complex Internet services interoperate, with the aim of improving resource usage (and therefore, containing operational costs) and performance – and in turn user experience and, arguably, profitability.

In practice, centralized configurations consisting of tightly coordinated software services (databases, computing nodes, web servers, etc.) have become a *de facto* standard for the delivery of Internet services; specific configurations and optimizations within this topology are the subject of scholarly research in the field of engineering and computer science, as well as experimentation and debate through engineering blogs[107] and specialized conferences[108]. Consequently, centralized configurations are deeply inscribed in most of the software designs and implementations that have been in common use throughout the past decade for the development of Internet infrastructure and services: redecentralization efforts, therefore, often face the additional struggle of having to design and implement not only the user-facing software and services that are the immediate products of their development efforts, but also considerable portions of software infrastructure on which to base their decentralized services.

Computational and information management problems that have been reliably solved over the past decades within tightly integrated, centralized configurations often need to be revisited within decentralized contexts; of specific relevance to decentralized Internet in everyday life are the fundamental issues of how data is stored, shared and accessed, and how identity and trust between computational actants (both humans and software) is managed. As distributed systems started being increasingly adopted in the second half of the 2000s decade[109], a better theoretical understanding of their capabilities and limits has been developing within the field of computer science: for the purposes of this chapter I employ the slightly simplified observation that as size, complexity and spatial distribution of computational infrastructures increase, the ability to rely on them as sources of coherent knowledge[110] provided with negligible delays depends on complex software engineering efforts as well as organizational and technical coordination.

In other words, decentralized (and distributed) projects inevitably face the challenges of providing computational services as conveniently and reliably as the incumbent centralized configurations. Because of typical limitations of home Internet connections, for example (as discussed in Chapter 4), fully decentralized data synchronization projects cannot reliably operate between multiple local networks (e.g. to keep a student's files synchronized between her computer or other devices at home and her smartphone while travelling or while connected to a campus network), as these devices are not able to directly locate each other and establish a data exchange connection without some form of mutual 'introduction', which in the case of centralized applications is provided by the vendor's server

[107] It has become common practice for Internet startups to publicly discuss software engineering processes, accomplishments (and, occasionally, failures) through *engineering blogs* to which employees operating in technical roles are invited to contribute.

[108] Amongts the most visible such conferences, the "Velocity" series of yearly events organized by O'Reilly Media is one of the specialized extensions of the "Web 2.0" conference/summit series (2004-2011, see Chapter 4), focused on operational challenges related to optimization of performance of Internet infrastructure.

[109] It is important to maintain an analytical distinction between *distributed* and *decentralized* computational systems: whereas the actual topology may be similar, consisting of a number of distinct computational nodes, *distributed* only relates to the physical topology, whereas *decentralized* implies the lack of a single coordination agent, except one that could be created through algorithmic consensus, such as the case of Bitcoin. In other words, *decentralized* systems are always also *distributed* to some extent, whereas *distributed* systems are not necessarily *decentralized*, and can actually appear to an external computational agent, such as a client computer, as a single server, whereby all the infrastructural complexity is 'black-boxed' behind user-facing server infrastructure.

[110] I.e. such that, *ceteris paribus*, the same request for a discrete piece of information or data submitted from any node of a large decentralized network results in exactly the same answer.

infrastructure, at the price of disclosing personal metadata (e.g. a user's location and travel patterns). Redecentralized projects address this class of issues by either providing some server-side coordination, with the ability for users to run their own coordination nodes or to choose trusted ones, or by using distributed databases similar to those employed by p2p anonymous filesharing applications such as Bittorrent, constantly updating information about the location of each peer and propagating this within the network of connected nodes, which – collectively – acts as a replacement of centralized coordination.

This strategy is used, amongst others, by the Kademlia p2p distributed hash table (DHT) protocol (Maymounkov and Mazières 2002), as a base layer upon which several higher-level software projects depend: although DHT protocols and implementations have been improved and optimized for different contexts and use cases since these early designs such as Kademlia, the issues of peer discovery and p2p data exchange that these address can be considered as essentially solved from a technical point of view. The social and organizational preconditions to their adoption as a common way to organize exchange of personal data in a secure and reliable way in everyday Internet use, on the other hand, continue to be a terrain of struggle for redecentralization projects, suggesting that trying to replace incumbent configurations through technical means only is an unsustainable strategy. Discussions amongst developers of redecentralization projects, as well as seminal scholarly research within this field (Wachs 2015), highlight how decentralized infrastructures depend vitally on the agency of end users: for example, through the ability to run independent coordination nodes on which family members, friends, co-workers, etc. (rather than peers negotiated algorithmically or inferred from an user's social graph on SNSes) can rely in the many cases in which running their own computational infrastructure is not feasible or practical, therefore relying, in practice, on mutual trust and social structures of support not based on more recent and often unfamiliar (as well as opaque) digital ways of modelling and mediating trust and social connections. These still unscrutinized digital ways may actually not only reproduce the hegemonic roles that decentralized configurations aim to liberate users from, but even intensify and amplify the opportunities for unsupervised and automated control over users (Feenberg 2002, pp101–113).

The social and political aspects of decentralized configurations have been the subject of study in the physical contexts of wireless community networks across locations worldwide (Flickenger 2003; Jungnickel 2013; Powell 2008; Powell and Regan Shade 2012; Vega, Cerda-Alabern, et al. 2012), as well as a topic of sustained interest for redecentralization projects, both focused on local contexts or on general principles[111]; as examples of tangible technical arrangements that rely on personal acquaintance amongst members of each local project, community wireless networks strive to address similar issues to the contexts discussed above, and highlight analogous challenges related to the actual democratic relevance of decentralized configurations beyond the technically competent groups (*geek-publics*: Powell 2008) who are able to appropriate and reinterpret such configurations more easily than *community-publics* (ibid.).

Similar considerations can be developed for the substantial challenge faced by redecentralization projects of providing practical and secure alternatives to centralized, corporate-controlled identity and trust[112] management systems, on which most personal Internet applications used in everyday life vitally depend: although theoretical models of computer-mediated identity management and trust have been well understood for over two decades and have been employed in widely available software implementations[113], their uptake amongst non-hackers[114] is negligible. Fahl, Harbach, et al. 2012; Ruoti, Andersen, et al. 2015; Whitten and Tygar 1999 identify key challenges in the ways in which the user interface of applications for encrypted communications and for management of mutual trust between individuals fails to make the underlying mechanisms of public-key cryptography and web of trust accessible to non-technical users. Recent projects such as Keybase[115] have tried to address these persisting issues by leveraging the social graph of mainstream SNSes: instead of relying on the ability of users to understand the security implications inherent in confirming their mutual identities whilst not able to meet in person, mutual trust is algorithmically composed by confirming several distinct 'proofs' carried through each individual's public statements on SNSes. This approach, however, as for the examples discussed above exposes the limitations of a technology-focused approach: firstly, existing

[111] Amongst the seconds, the Electronic Frontier Foundation (EFF) has been considering open wireless networks a priority focus since 2011 (Open Wireless Movement 2011) and the OpenWrt (`https://openwrt.org/`) and CeroWrt (`http://www.bufferbloat.net/projects/cerowrt`) projects produce free software open firmware for home wireless routers that can be used in mesh configurations, to share wireless Internet access with neighbours or for other ad-hoc p2p traffic routing configurations.

implementations are proprietary and controlled by single corporate entities, as well as relying on incumbent, centralized and proprietary SNSes; moreover, and most importantly, they are based on the assumption that social and cultural norms can be effectively circumvented through computational means, effectively bypassing the challenges inherent in a fully democratic reimagination of the fundamental layer of identity and trust online, outside of the technocratic rationalizations on which most of current Internet services rely.

8.4.3 Agency and computational capacity: the contested execution of code

As mentioned at different points throughout this chapter, decentralized alternative rationalizations inherently require the ability for users, or groups of users, to run software code that would otherwise be run within corporate datacentres when using centralized mainstream services. Developing code that responds to user needs through decentralized topologies, in other words, is only part of the problem: code needs to run somewhere—ultimately on physical hardware, even when this is abstracted through computational capacity made available as 'cloud computing'. Execution of software code is therefore part of the computational complexity that alternative rationalizations have to deal with, and it involves in turn issues of control and power. On one hand, common end users cannot be expected to deal with the complexities of operating what is essentially 'datacentre infrastructure': not with regards to its physical location but to the operating principles, which are different from those of consumer or office systems. Infrastructure nodes are supposed to be constantly on and connected to the Internet, to operate without manual supervision for most of the time and to be kept up to date without user intervention; moreover,

[112] A trend emerging in recent years is the reliance of websites and web/mobile apps on authentication and authorization (regulating access to information, resources, etc.) dependent on a user's identity on the Google, Microsoft or Facebook platforms, even in contexts operated by unrelated organizations. Decentralized and open identity systems with free software implementations have been available for over a decade (e.g. OpenID), but although they are widely used as infrastructure for authentication, their use as fully decentralized infrastructure (e.g. operated by privacy-focused organizations or individuals) is negligible.

[113] For example, the 'web of trust' model at the base of public-key encryption software such as PGP and successive free software implementations of the OpenPGP specification, such as the GNU Privacy Guard (GPG).

[114] And amongst hackers as well, partly as a side effect of low network value of such tools because of sparse overall use except for very specific security contexts.

[115] `https://keybase.io/`

datacentre infrastructure is normally operated without any screens attached to it, therefore making the kinds of visual checks common while troubleshooting malfunctioning of consumer devices impossible: management of systems is routinely carried out via remote connections, often using command line interfaces that most common users would not be familiar not comfortable with. On the other hand, most of the projects I surveyed rely on the assumption that as execution of code is allowed to be more detached (physically and in terms of control) from individual users, the least control these users can exert over it, as this separation may involve different kinds of compromises; for example, relying on servers in a data centre does not offer—in most jurisdictions—the same level of protection from arbitrary interference from third parties: a data centre operator voluntarily disconnecting an user's server or being affected by network outages, police seizing hardware without the same kind of warrants they would need to enter a private dwelling, etc.; relying on commercial operators of otherwise interoperable, decentralized and free software alternatives, likewise, may offer some additional convenience but similarly forces users to rely on organizations whose ultimate goals may not be aligned with those of their users.

In response to such concerns, several projects have been attempting to create computational environments[116] that combine software and cheap, physically unobtrusive hardware that consumes little power, such as the Raspberry Pi 'system–on–chip' (SOC) computers, with the aim of creating small personal servers that can be operated without any advanced technical skills, on which common users can run lifeworld Internet services for themselves and for their family members or friends, and that can communicate directly with similar systems operated by other individuals. Personal servers constitute an ongoing domain of development which has so far enabled committed individuals to experiment with alternatives that allow end users to retain control not only over code and content but also over execution of code. Interestingly, not only projects sustained by political motivations (whether based on volunteer work or incorporated as a business) have been exploring this domain, but also mainstream consumer electronics companies, such as vendors of hard drives and storage solutions: albeit a commercial niche, personal storage and application servers that rely on proprietary software (in turn often based on free software components) are being commonly sold in consumer electronics shops and similarly to alternative rationalizations provide features that let home users manage media, backups and data synchronization within their home network; these devices, however, are typically provided as black boxes that only allow limited reconfiguration, while the vendors ultimately control which code can be run on them and for how long.

Other, very recent projects, such as Ethereum[117] —whose motto is 'a decentralized software platform'—and Nymote[118] —whose motto is 'Lifelong control of your networked personal data'—are investigating strategies that could provide distributed computational capacity through simple hardware or cloud nodes that don't run any specific software services (as is the case with mainstream Internet server and also with the personal servers discussed above) but only provide the ability to execute code in a p2p way, whereby not only data or content (as in the case of p2p filesharing) but also code is distributed throughout the network and executed according to available capacity. Although a potentially interesting strategy for specific contexts, the practical utility of these projects for alternative rationalizations cannot be assessed until they are more established: they are mentioned here specifically to highlight the additional epistemic complexity involved in strategies for the execution of code that further abstract the idea of what software itself is, when distributed in a p2p fashion similarly to content, and that further deconstruct the idea of software services by reducing them to mathematical functions that can individually be run on any node of a computational network, whilst collectively providing a user-visible service. These abstractions visibly challenge current mainstream representations by making the exact location and the meaning of code execution even more opaque than in cloud services, which in turn raises concerns about control and ownership of computational capacity that future research will likely have to explore.

To conclude this brief overview of challenges and strategies involved in the control over execution of code in decentralized configurations, it is worth returning to a working assumption that I introduced in Chapter 4 while discussing the increasing adoption of mobile devices. Whereas my earlier discussion highlighted how these are powerful gen-

[116] For example, the FreedomBox project: `https://freedomboxfoundation.org/`, besides several crowdfunded commercial projects.

[117] `https://www.ethereum.org/`.

[118] `http://nymote.org/`.

eral computation devices that most people in affluent countries carry with them in daily life, when looking at the details of how these classes of devices (smartphones, tablets) provide computational capacity, pervasive issues of control become manifest, making the use of these devices as personal servers independent from commercial entities substantially unrealistic beyond ad-hoc configurations unsuitable for large scale adoption. Technically, any recent smartphone or tablet could provide ample computational and storage capacity to run personal services: as these devices can typically be constantly connected to the Internet, even when a user is travelling, they could allow to avoid the technical limitations of domestic networks that make access to personal servers within home networks difficult from outside—a user could always carry with them a copy of their content and data, synchronized with the 'canonical copy' stored on a personal server at home whenever getting back to the domestic environment. However, smartphones and tablets—differently from laptop and desktop computers—almost invariably require specific configurations of their operating system: whereas most consumer and office computers can be easily exchanged with a more recent model, for example, on which the same operating system and applications used on a previous one can be installed, mobile devices are normally locked by their vendors so that no other versions of the same operating system can be installed (for example, a version of the Android OS that does not intimately rely on Google services, such as the Cyanogenmod project[119]) even when the actual software code of the alternative version of the operating system could—technically—run without any issues. Furthermore, even when devices can be successfully unlocked (either as allowed by the vendor or through hacks that exploit bugs in the operating system originally installed), any replacement operating system needs to be built for the specific model and version of mobile device on which it is going to be installed: as most of these devices rely on specific hardware components (data modems, sensors, cameras, power management, etc.) for which no generic 'drivers' are available (as is normally the case with desktop operating systems), users cannot simply install (for example) a generic version of the Android OS that does not force to use Google services on an arbitrary device that is sold with the Android OS installed by its vendor: each device requires considerable efforts in terms of customizing and testing a device-specific version of the Android OS, making longer-term support of alternatives to vendor-provided software unsustainable. This ultimately forces users to depend on software that they don't control and that they can't reliably appropriate and reconfigure in order to use their mobile devices as generic computational devices.

As for hardware for personal home servers discussed earlier, the ability to control code execution on mobile devices is an increasingly relevant terrain of struggle for alternative rationalizations: recent hacker discourses are visibly highlighting the stark contrast between the ample computational capacity commonly available at low cost, in convenient physical formats and requiring low power use, on one hand, and the ultimate inability for common users—and often for committed hackers as well—to effectively use this computational capacity to execute arbitrary code outside of the boundaries set by hardware producers.

8.4.4 Political economy of redecentralization

Having explored challenges faced by hackers involved in redecentralization projects in the previous sections, I finally turn to analyzing the emerging complexity of the political economy surrounding decentralization efforts. Allocation of suitable resources to the development efforts required by alternative rationalizations may appear in principle to be made increasingly easier by the availability of relatively cheap and flexible computational capacity, as discussed in relation to the computational turn of the Internet in Chapter 4; on one hand, however, computational capacity is only part of the resources needed for any development project: the labour of contributors, whether volunteers or salaried, is typically the core asset of each project, and its availability is affected either by the ability for volunteers to dedicate time to a project thanks to existing reliable income through their day jobs or by the ability to directly remunerate salaried contributors. On the other hand, whereas computational capacity (e.g. cloud servers and storage) needed for the development of software projects may be easier to acquire when large upfront capital expenses can be avoided, for example by running code on cloud infrastructure rather than purchasing and managing own hardware, access to it naturally still involves some form of substantial expenditure, especially as projects grow to accommodate increasing numbers of users

[119] `http://www.cyanogenmod.org/`.

through their infrastructure.

Free software projects are often reported as struggling to sustain their development efforts through adequate funding; even projects producing software on which essential parts of the global Internet infrastructure relies are often run by volunteers supported by meagre donations, as became evident when a major software vulnerability ('Heartbleed') was discovered in April 2014, affecting the OpenSSL cryptographic software that is used pervasively in a large part of free software and proprietary Internet code that relies on cryptography features: the project's treasurer described (Marquess 2014) how their funding amounted to 2,000 USD *per year* on average through donations, with additional funding being procured through consulting, concluding his review by highlighting: 'So the mystery is not that a few overworked volunteers missed this bug; the mystery is why it hasn't happened more often.' Only the unusual public attention around this software vulnerability disclosed how even large companies were benefitting from its use, while failing to make sure that the project was appropriately funded and able to sustain the amount of complex programming work needed to ensure that reliable code reviews could be performed: shortly after, several of these companies pledged to contribute to a fund (the 'Core Infrastructure Initiative') managed by the Linux Foundation and meant to support free software projects used throughout the Internet infrastructure and struggling to attract sustainable funding (Linux Foundation 2014). Other key free software is similarly funded through the contributions of companies who rely on it: breakdowns of code contributions to recent Linux kernel releases analyzed by the news site Linux Weekly News typically show that more than half of the code contributed to each release is written by employees of large technology companies such as AMD, Intel, RedHat, Samsung, etc. (Corbet 2015).

Apart from a few free software projects that can rely on direct or indirect corporate sponsorship, however, reliable funding able to attract sustainable development work and to cover any operating expenses is a core issue for independent projects. A trend that became visible in the second half of the 2000s decade is that of crowdfunding of free software projects. Whereas most such projects have traditionally been open to donations by displaying on their websites invitations for users to support the project through occasional or recurring donations, crowdfunding operates in a different way, by establishing an upfront agreement between a project and its potential users, who are invited to pledge towards a minimum amount of money that is considered by the project as necessary to commit to developing an agreed set of features. Although forms of funding of independent projects based on contributions from users, fans or donors are not a new phenomenon (emergency response funding appeals largely predate the Internet, for example, although running these online can allow to gather funds more quickly in emergencies), recent online crowdfunding platforms such as IndieGoGo and Kickstarter have managed to establish crowdfunding as a popular funding model for technology projects, by abstracting the technical, computational and regulatory details needed to operate crowdfunding campaigns into online services that connect projects to funders (crowdfunding platforms can therefore be seen as a specific form of Software–as–a–Service enabled by the computational turn of the Internet, relying on a share of the amounts raised by projects as their own income).

Whereas scholarly literature on crowdfunding has been developing so far mainly in business and management journals, focusing on business models, patterns leading to success or failure of funding campaigns and on the delivery of funded products or services to funders (e.g. Kuppuswamy and Bayus 2014; Mollick 2014), in the context of alternative rationalizations of lifeworld Internet it is useful to focus on more intimate links between developers and funders: Belleflamme, Lambert, et al. 2014 compares '[...] two forms of crowdfunding [...]: pre–ordering and profit sharing. In the first form, entrepreneurs invite consumers to pre–order the product, to collect the necessary capital for launching production. [...] In the second form of crowdfunding, entrepreneurs solicit individuals to provide money in exchange for a share of future profits or equity securities. In this profit–sharing scheme, the investors may or may not decide to consume the product at a later stage.' (ibid., pp585–586). In the specific case of free software (or 'open hardware') projects such as those analyzed earlier in this chapter, however, the model of synergies between funded projects and their funders often involves both an immediate utility (the ability to use the code produced) and a peculiar, non–monetary form of profit–sharing through the ability to reuse free software code in derivative projects. Additionally, due to the free software license terms, anyone—including individuals and organizations who didn't contribute to the funding—are allowed to use the software developed as part of a campaign.

Moreover, campaigns that successfully raise funds but eventually fail to deliver the promised software or hardware in the form that was agreed with funders often still contribute indirectly to the understanding of struggles involved in the open development of software or hardware projects that aim at replacing mainstream affordances: public write-ups of both successful and failed crowdfunded projects are often discussed at length in hacker circles, especially when providing honest and useful insights into the challenges involved in delivering on small budgets alternatives to products and services that can rely on large pools of resources within mainstream corporations.

Recognizing the importance of exploring ways of connecting hackers and users beyond the plain economic exchanges of traditional crowdfunding, peer–to–peer crowdfunding projects have been created recently; one of the earliest such projects, VODO[120], is actually focused on indie culture rather than on software projects, but through personal discussions with its founder and one of the initial contributors I was first introduced to the idea of a peer–to–peer computational system to directly match 'viewers' (as the project was initially focused on independent filmmaking) and artists:

> The system [...] includes a table of hashes of files (initially as contained in .torrent files) that are out there on P2P networks against payment information for creators. We encourage artists to submit their hashes and payment details to VODO in order to build a lookup table of donation information that can then be propagated out through various sites, services and platforms. [...] [w]hen you make a donation through VODO you're not paying either for an 'illegal download' of a work. You may or may not have downloaded a work. All we know is that you have arrived at VODO from somewhere with a VODO link—which can be a button on a page, a text link, a link on a tracker, a link on an artist's page—intending to make a donation to an artist. You make the donation to them directly, not in respect of any particular use of their work, but simply to support them directly. They may have licensed their copyrights to some third party, but you can still support them directly, if they allow you to do this. For example, you might buy the Radiohead album but still decide to send them $5 because you love them so much. (King 2008, personal communication)

By making it possible for fans to send money directly to authors, the VODO project aims at circumventing the layers of intermediate bureaucracy and gatekeepers represented by the incumbent media industries: acknowledging the practices that several of my field-

[120] `http://vodo.net/`.

work participants discussed—that downloading cultural works via peer–to–peer networks is 'an emerging norm'—the computational matching of digital fingerprints of works, authors and fans operated by the VODO project created over the intervening years a niche but successful system of funding that is part of a political economy in which the role of the media industries as cultural mediators is being *de facto* made largely redundant by a public of consumers who can easily discover and procure digital artifacts through a range of human (e.g. bloggers, peers) or non–human actants (e.g. algorithmic recommendation systems).

Within the context of software alternatives to mainstream Internet affordances, more recent projects such as Snowdrift[121] are exploring peer–to–peer funding models that—similarly to traditional crowdfunding—allow hackers to seek financial support for their work by agreeing features directly with their funders, while removing the need to rely on commercial crowdfunding platforms: on one hand, the authoritative role and the 'brand value' of the major crowdfunding platforms may be seen as becoming less relevant as hacker publics are increasingly familiar with the crowdfunding model; moreover, except for the processing of payments—to which peer–to–peer alternatives to credit cards processors such as bitcoin and other digital currencies are only a limited alternative—the computational complexity of crowdfunding operations is considered minimal[122] and it has in fact been successfully replicated by several free software projects, both for their own use or as platforms through which others can run crowdfunding campaigns independently.

More importantly, however, the conceptual model of Snowdrift is based on a closer relationship between funders and hackers: rather than simply coming across interesting projects that they may wish funding, individuals are invited to consider which amount of money they would consider fair (according to their own use and financial situation) to contribute to free software projects, and to allocate their budget to specific projects they consider worth supporting, therefore becoming direct stakeholders on a longer term rather than simply contributing one-off donations. Interestingly, Snowdrift is formally a co-operative organization and is openly based on principles of cooperation. Although Snowdrift is still running as an experimental project, its proposed model seems to match at the level of political economy the 'community model' of development of the Internet discussed by Feenberg (Feenberg 2012), in a way exploring strategies to reconfigure the computational agency provided by crowdfunding platforms to better match the needs of hackers involved in alternative rationalizations and in turn the computational agenfy of the users of these alternatives: whereas the success of these funding models can only be assessed by future research, they nevertheless represent original attempts to reshape hackers' own representations of how funding for independent projects can be imagined, through strategies that try not to rely on traditional economic assumptions of capitalist funding strategies.

A further experimental model of funding of alternative rationalizations, focused on making the costs associated to running decentralized infrastructure sustainable (rather than on funding of development efforts) was briefly mentioned in the earlier section on IndieWeb; whereas free mainstream Internet services rely on advertising or on monetization of user data and attention to sustain their operating expenses, funding of the operations of decentralized infrastructure needs to take into account the multiplicity of actants involved, some of which may operate as commercial entities, while others may rely on reciprocal exchange of services. Companies such as IndieHosters (discussed earlier) charge their users recurring fees in exchange of technical support and advice, rather than relying on the rent seeking model of traditional Internet hosting services. Other projects, such as Sandstorm[123], CozyCloud[124] or arkOS Connect[125] provide either plain computational capacity that users can configure as needed by activating a range of lifeworld Internet services (private communication, storage, etc.) or 'managed' computational capacity, whereby technical support, advice and consulting are offered to users in order to help them maintain their private services. Interestingly, all these (commercial) services were initially started as development projects that focused on alternative rationalizations to mainstream lifeworld Internet services, upon which a commercial offer was added at a later stage: any expert user, therefore, could install on her own infrastructure the software code devel-

[121] `https://snowdrift.coop/`.

[122] A popular online course run in 2013 by the Stanford University ('Startup Engineering') included as a practical coding assignment the development from scratch of a crowdfunding platform, which students were then invited to use to attract real funding for the startup plan they would develop throughout the course, highlighting in a tangible way that the technical mechanics of crowdfunding platform software are considered substantially banal from a computer science perspective.

oped by these projects and on which their commercial offers rely; in turn, this computational capacity can either be operated for personal use, for family or friends or offered on a commercial basis, therefore making alternative rationalizations available to others while ensuring that the costs associated to operating the underlying computational capacity are sustainable. As for the collaborative funding platforms just discussed above, the actual success of these commercial enterprises can only be assessed over a longer timescale, but they nevertheless represent visible attempts to imagine a sustainable way of running decentralized infrastructure.

Although most of the alternative funding models discussed in this final section are still experimental, only operate on small scales—even when they often appear in hacker discourses (and in several cases in mainstream tech news sites)—and have not yet been extensively validated by scholarly research, I nevertheless chose to briefly discuss them here because even a tentative sketch of elements of a political economy of decentralized Internet infrastructure can start to identify actual attempts, operated by hackers, to seek viable ways to sustain the development and operation of alternative rationalizations, addressing the key issue of physical control over the execution of code—discussed in the previous section—that currently constitutes a substantial limitation to the computational agency of users and hackers even when suitable free software code is available to them, while still needing to be run reliably.

8.5 Conclusion

Throughout this chapter I have attempted to develop an analysis of computational agency in lifeworld Internet through a close reading of the material role of software code, of the development practices and of the hacker motivations involved in the imagination and production of alternative rationalizations. On one hand, the analysis has been focused on specific contexts that are relevant to the challenges discussed while analyzing my own Web 2.0 project (Chapter 5) and to the user accounts discussed in Chapter 6. The IndieWeb move-

[123] `https://sandstorm.io/`.

[124] `https://cozy.io/`.

[125] `https://connect.arkos.io/`.

ment, for example, through its focus on the individual and on ownership of and control over personal content, constitutes an example of practical implementation of the principles of self-determination that were embedded in the design of my 3D Graduate application, and the IndieWeb's POSSE/PESOS strategies can be seen as a successful—even if for a small group of users—execution of my attempt to let students make sense of content from heterogeneous sources. The decentralization strategies discussed in the second section of the chapter, likewise, constitute attempts of implementing in practice configurations of software that more closely match the expectations that some of the students I interviewed discussed, when they articulated their struggles to meaningfully fit within their everyday life the technical form of mainstream services that require the constant mediation of corporate entities whose accountability was often questioned.

On the other hand, whereas the actual uptake of most of the alternative rationalizations discussed in this chapter has been shown to be almost negligible if compared with the global scale of major social network sites, the ability for independent hackers or groups of hackers to build software that can reliably replace parts of mainstream lifeworld Internet even for a small subset of privacy advocates or users concerned with the centralization of control over Internet services represents a remarkable achievement when considering the uneven distribution of power between mainstream corporations and independent developers. On one hand, as capitalist agendas—either through large advertising/Internet corporations or through the Silicon Valley–style startup culture—increasingly shape mainstream Internet, the efforts of hackers who dedicate energies to alternative rationalizations draw attention to the role that critical thinking can have in questioning in visible ways the values that are inscribed in the technical materiality of mainstream Internet infrastructure (Berry 2014): technical determinism and a one-dimensional telos of progress are not only critiqued theoretically, but also shown in practice not to be an unavoidable way of dealing with technology in the sphere of everyday life, but part of complex power struggles that while also including technological determinism(s) are nevertheless increasingly affected by the visibility of a very heterogeneous hacker public that enacts contestations and negotiations over the role of software code and of hacking (Jordan 2009). Secondly, as the struggles analyzed in the last section of this chapter indicate, contestation of code and through code requires not only the ability to develop sustainable alternatives and to address the limitations imposed by the technical materiality of mainstream Internet, but also to establish close connections with the everyday life of common users, which has been so far an elusive domain for independent projects: whereas large companies routinely employ anthropologists and dedicate resources to user research, this is very seldom possible for independent hackers. Although a common source of motivation for independent projects has traditionally been the need to serve the specific needs of their own developers (Coleman 2012a, chapter 3), the ability for the hacker recursive public to increasingly include non-technical users in discourses about alternative rationalizations could provide additional user feedback by further blurring the boundaries between common users and hackers at least in the intimate context of lifeworld Internet. It is in this sense that hackers have been analyzed throughout this chapter as cultural and computational mediators: not only as expert hands writing code that common users would not know how to write and operating complex decentralized infrastructure that common users would find obscure when compared with the convenience of mainstream Internet services, but also as a public that—as far as it is successful in enrolling common users—can gain direct visibility into the everyday concerns of users intent in finding a place for Internet technologies in their lives, thereby acquiring the ability to incorporate their concerns and their strategies in further developments of alternative rationalizations.

Chapter 9

Conclusion

Throughout the dissertation's empirical chapters I sought to explore how computational agency is configured within the domain of lifeworld Internet, across the transition towards the computationalization of Internet infrastructure, developer practices and user-facing affordances that I labelled as 'computational turn of the Internet'. The analysis started from the lifeworld of a small group of users in a local context and my original plan was to investigate how users would approach reconfiguring and appropriating affordances designed to incorporate malleability (Web 2.0 apps); however, my approach highlighted that in order to understand user agency it is necessary to transcend and question the spaces for user choice and action codified in software, and to trace agency downwards through the multiple layers of technical infrastructure and developer labour that progressively shape affordances as delivered to users.

In other words, the progressive potential of Web 2.0 didn't happen not only because the Web 2.0 narrative was articulated as a business-oriented manifesto that ultimately addressed the interests of capitalist accumulation of value and rent-seeking by extracting value from users and concentrating it under the control of private entities: counter-narratives to hegemonic agendas did certainly exist during the early years of Web 2.0 — as they have through the voices of hacker counter-cultures since the very beginning of the Internet — but knowledge, strategies and 'tooling' (constituting part of what I call computational agency) were not able yet to sustain actual development of alternative rationalizations that could effectively contest the dominance of mainstream affordances on an even ground. The constitution of a new global recursive public of hackers able to act as computational mediators was framed in the final empirical chapters as the main factor that allowed the potential progressive uses of computational capacity to attain realization through hacker labour and reproduction of hacker knowledge and concerns.

9.1 Review of research findings

The first empirical chapter examined the constitution of mainstream Internet through an analysis of its technical materiality: through this, I developed the essential conceptual framing underpinning the analysis of the later empirical chapters. Firstly, I discussed how the architecture of recent lifeworld Internet is intimately connected to two-ways flows of information and to context-specific spaces for the reconfigurability of affordances; although this is rooted in early Web 2.0 narratives, actual later configurations can be deconstructed to show the increasing relevance of non-human actants, complicating the Web 2.0's focus on human participation and highlighting how machinic agency is deeply interwoven with human agency, firmly representing within the network of participation the interests of those that control the non-human, computational actants. I further deconstructed the assumptions and historical legacy of the leading topology of Internet architecture (a client/server one), highlighting how this legacy has contributed to its transition to hegemonic role even in the many cases where a peer-to-peer, distributed topology could be technically more efficient or democratically preferable, whereas decentralized and peer-to-peer alternatives are relegated to subaltern status. Finally I articulated the fundamental concept of *computational turn of the Internet*, that I then used in later chapters to identify implications on user and developer agency of the precipitous shift happened around the year 2010 towards a much higher reliance of all the layers of Internet infrastructure and user-facing affordances on computation than at any previous stage. Anticipating the contestations of hegemonic configurations discussed in the final chapters, I deconstructed four key traits of the computational turn of the Internet highlighting how they had until then mainly been seized by large Internet corporations to develop and sustain large scale proprietary SNSes and Internet services, while being potentially open to being used by independent developers interested in alternative rationalizations.

Throughout the second empirical chapter my focus was the review of the process through which I developed an early Web 2.0 application, based on an user-centric design that was partly developed through Web 2.0 strategies involving content mashups: although ultimately this app did not have a real direct impact on users' computational agency due to the fact that the project was not completed in its original form, looking at the negotiations between competing agendas, at the assumptions about actual use of Web 2.0 strategies within the app's public, and at the technical issues involved in the software engineering processes of early Web 2.0 applications, I traced — within the context of my case study — the network of power relations that complicated the optimistic assumptions about user choice and agency proposed by the Web 2.0 narrative.

I then returned to the same fieldwork site for a further phase of fieldwork in 2010: my third empirical chapter deconstructs user experiences alongside three main analytical domains. Firstly, I traced the practices emerging from the accounts of research participants to the technical form of earlier Web 2.0 applications and to the associated narratives; my analysis highlighted that the specific material implementations of Web 2.0 traits had not attained any significant uptake amongst my reference users: whereas the *goal* of facilitating production and sharing of content had been realized (practically all the students engaged in significant practices of content production, in very personal forms), the tools through which this was being done had little resemblance with the 'composable building blocks' shape through which Web 2.0 affordances could be described, and I did not encounter any significant use of automated aggregation of content: 'mashups' strategies were employed by these students, but largely through manually curated interventions. Secondly, I explored how user practices had been reconfigured through these students' initial exposure to computational web environments such as SNSes: I found that appropriating new technical form (for example, learning how to use peer-to-peer filesharing applications to find and procure content) was generally done without any significant hesitation, whereas most students were still trying to make sense of the reshaping of social norms faced through the daily use of lifeworld Internet affordances, highlighting how *operating* technical affordances was generally not seen as a problem, whereas *making sense* of their role in everyday life implied more delicate, personal learning processes. Finally, I articulated the deconstruction operated by a few research participants of their own lifeworld Internet: these students revealed various degrees of confidence in their ability to meaningfully reconfigure and appropriate technical form beyond what inscribed in affordances available to them, and connected their practices, which involved the use of free software and the attempt to subtract their lifeworld Internet from the centralized control of mainstream Internet corporations, to clear political concerns about privacy and computational agency.

In the two final empirical chapters I followed the networks of power relations inscribed in technical form as they had emerged from my local fieldwork, in order to identify whether and how these are being contested and what the implications of contestation on computational agency is.

In Chapter 7 I described the progressive constitution of a new recursive public of hackers, analysing how their recursive role has allowed them to establish themselves as a public able to take part in negotiations over agendas being inscribed in technical form, able to reproduce itself recursively through peer-induction operated via discursive strategies and via computational strategies (learning to write code by reading others' code, engagement in gamified learning environments designed to foster quality contributions, employing computation to promote learning of computational thinking), and able to sustain self-reflective discussions about hackers' own role as cultural and computational mediators in a world increasingly dependent on computation.

In the final empirical chapter I focused on alternative rationalizations of technical form within two specific domains connected with practices, implicit struggles and explicit political debates I observed through my local fieldwork. The first domain analyzed is that of strategies focused on ownership/control over one's own content online, with the aim of avoiding the restrictions of corporate information silos, while still being able to engage in meaningful conversations with friends, family and other contacts who prefer to use mainstream SNSes; one specific trait that emerged as relevant from hacker discourses, and that is related to the previous chapter's analysis of the constitution of hackers as a recursive public, is the importance of peer-support and induction that is a fundamental part of the ethics of the hackers involved in IndieWeb development. The second domain analyzed is that of strategies of decentralization, aimed at subtracting computation, data storage and transfer from the centralized control prevalent in mainstream Internet as discussed in Chapter 4. Finally I reviewed the struggles experienced by hackers while developing and promoting alternative rationalizations in an historic juncture characterized by increased centralization of control: user engagement, efficient and meaningful management of convenience and trust, and physical ability to execute code and to control computation are open issues that are affecting the ability of alternative rationalizations and hacker counter-narratives to establish themselves firmly within public discourse.

9.2 Answering the research question

If we look back closely at my research findings through the letter of this dissertation's research question, evaluating user agency within computational lifeworld at the different stages of my research, the answer is different at each stage. When considering the students that should have been the users of my 3D Graduate web app, there could actually be no conclusive answer, as the app was never used in its form as originally designed by myself around Web 2.0 principles of integration of different sources of content, except for early versions that were tested by a couple of dozen students. These students' responses were substantially polarized: some commented that they didn't see the point of the 3D Graduate app, whereas the few who happened to be enthusiastic early users of Web 2.0 apps in their everyday life signalled that they would be eagerly waiting for the final version to be made available to students; as the app's development was stopped and as the platform finally delivered to students had a much less ambitious shape, however, there was no real impact on users' computational agency within the scope originally set. On the other hand, however, the complex failure of the project and its ultimate failure to provide students with a Web 2.0 tool through which they could meaningfully reflect on their personal development highlighted the problematic lack of effective alliances between key actants (myself as lead developer, developers of software my project relied upon, and the students as end users) that could have steered the project in a different direction. The likelihood of this specific project to succeed would realistically have been very low nevertheless, as much of its technical materiality, given the scarce resources available, depended on the ability (or lack thereof) of FLOSS software available at the time to effectively incorporate concerns of individual users into Web 2.0 architectures; the crucial lack of alliances was, instead, more systemic, as the successive corporate takeover of Web 2.0 spaces and of lifeworld Internet itself would highlight. My argument, essentially, is that the liberating potential of contestable technologies based on software code and of direct user engagement (Feenberg 2012) was not successfully seized to create spaces for users' computational agency because technical knowledge and capacity were not available at the time to form useful alliances with end users who, on their part, were unavoidably still struggling to understand which role the Internet could have played in their daily lives.

When looking at the practices of the students I interviewed during the second phase of my fieldwork, instead, we can see how by 2011 social network sites had successfully established themselves as a common fabric of users' everyday life, concentrating within consistent user interfaces multiple distinct affordances (chat, selective sharing of content, news feeds, games, a variety of niche apps) that students used for personal tasks or to collaborate with others (for example, to discuss coursework outside of the 'official' learning environment provided by the university). A general familiarity with how to create, share, procure content on the Internet was also shared amongst all the research participants, whether within the walled garden of corporate SNSes or through web applications that students would discover from peers, friends, family, and would generally be able to use with little effort — only occasionally enlisting 'warm experts' (Bakardjieva 2005) for initial help. User agency within lifeworld Internet had therefore been visibly transformed, at least in the context of my fieldwork, and for most of the students interviewed some combination of Internet affordances gravitating around a main SNS (Facebook for most of my research participants) was indeed part of their normal way of dealing with everyday tasks ranging from production and consumption of content to keeping in touch with friends and family. The few students who discussed their interest in exploring affordances not encumbered by the centralized control of private corporations contributed further useful insights over the ongoing reconfiguration of computational agency: while discussing quirky personal remixes of free software installed on personal cloud servers, 'rooted' mobile phones used as ultraportable computational devices, infrastructural interventions (through the use of VPNs or by setting up private VoIP systems), on one hand they highlighted that they were able to appropriate the traits of computational complexity that at the same time were sustaining the fast growth of proprietary SNSes and services, and on the other hand they positioned themselves within the ongoing hacker debates about centralization of computational capacity and about development of alternative rationalizations. The specific fieldwork context needs to be taken into account when assessing this successful engagement with hacker discourses: not only were these students a minority of my small group of research participants, but they were also members of the student community of a well regarded intellectual centre within a global city; the point, however, as discussed in my methods chapter, is not to establish statistical significance, for which my fieldwork methods would be unsuitable: what is relevant to my research aims is instead the successful engagement of research participants with practices through which, to various extent, their everyday life was augmented by computational capacity, while identifying different ways to relate to the growing concerns around colonization of the personal sphere by large corporations, ranging from uncomfortable acceptance to active refusal and contestation.

Finally, revisiting my research question at the time of wrapping up my research, in 2015, the answer is again complicated: on one hand, the focus of my later enquiry was on the hacker interventions through which computational agency was being reconfigured and not on end users; effective adoption of alternative rationalizations by common users is actually a pressing issue for hackers involved in redecentralization efforts, as discussed in Chapter 8. On the other hand, if we look beyond the context of my local fieldwork, reconfigurations of computational agency sustained by alternative rationalizations are increasingly visible: whereas most of the 'early adopters' of alternatives to mainstream Internet affordances are hackers themselves, the increasing visibility of the recursive public of hackers on which Chapter 7 was focused is effectively promoting a narrative of participation to the hacker public itself: on a global scale, web literacy initiatives such as the ones organized by the Mozilla foundation, besides their actual numerical relevance, promote very visibly (also through occasional appearances on mainstream media) the hacker narrative of appropriation of Internet affordances and infrastructure by fostering hacker learning focused on topics accessible to a general public. My own participation in an IndieWeb hacker gathering only a few months ago (July 2015) allowed me to experience directly how hacker narratives are successfully finding unexpected allies in common users curious about alternatives to mainstream options: rather than the uniform crowd of male geeks I would have expected to meet only a few years ago at an openly technical meeting, I encountered a mix of experienced hackers and passionate newcomers, often without much technical background and expertise at all, but eager to learn how to set up their own website, registering their own domain name, and experimenting with technologies that could complement the array of mainstream ones they had been using so far. Whereas obviously none of these examples could be representative of wider trends, they show that at least in some contexts hacker narratives are succeeding in engaging the interest of common users, progressively including them in the hacker public itself, through a mix of computational strategies, compelling narratives and peer induction.

9.3 Suggestions for future research

One core limitation of my research, due to the scope of my work, is the focus on mainly English-speaking hacker discourses: further research that could follow both users and hackers across cultures and languages could provide precious insights over the intimate relationships that I expect to observe between cultural contexts and approach to lifeworld concerns, including computational mediation.

As my evaluation above of the different answers to my research question through time and in different contexts suggests, longitudinal studies that could follow ethnographically large sets of users through time would also provide the ability to meaningfully compare individual users' approach to appropriation of lifeworld Internet at distinct time junctures, linking these to broader trends such as shifting mainstream configurations of Internet infrastructure. Whereas my research methods and the focus on computation as the layer of mediation between the social and the technical allowed me to connect the research findings of my local fieldwork to global discourses and transitions, for much of my later fieldwork focused on hacker discourses I often found myself eager to be able to rely on more established scholarly research that could help to clarify quantitatively the extent, velocity, articulation and relevance of the phenomena that I was observing as taking shape through the shifting 'Zeitgeist' of public hacker discourses: the little existing literature that could illuminate specific aspects of my enquiry (for example, quantitative studies of content quality on popular hacker forums for hacker learning and discussion) is cited in my final two chapters, but a much broader and varied set of primary research would be able to clarify in more detail the trends that appear from discursive analyses.

Although multidisciplinary research including quantitative methods could be useful to analyze specific aspects of computational agency in lifeworld Internet, the essentially private character of each user's lifeworld makes the analysis of each individual context necessary, exploring in depth both practices and justificatory discourses that relate practices to convictions, personal approaches to life situations and challenges, and to each individual's complex and often fascinating life stories; although these extend well beyond the layer of technical alliances, they connect back to these, as I observed almost invariably throughout my student interviews, shaping in unique and subtle ways each person's way of appropriating Internet affordances and of dealing with the exogenous limitations inscribed in these affordances. In other words, slow and patient ethnographic fieldwork would always be needed to unearth the irreducible traits of individual stories, beyond trends and broad categorizations that could be explored through popular quantitative methods such as sentiment analysis, analysis of behavioural data and of large corpora of textual data.

Moreover, individual lifeworlds are difficult to access — at least to independent scholars — on a large scale: available public data can provide useful insights but its use needs careful methodological considerations in order to account for all the missing data voluntarily kept private, and for the biases deriving from different abilities for different groups of users to use devices and applications and consequently to leave public traces that could constitute entry points to their lifeworlds for researchers. Researchers working with major SNSes have access to much larger and complex datasets than those available to independent researchers (e.g. Kramer, Guillory, et al. 2014), and can therefore engage in different kinds of analytical approaches, although the confidentiality of the data used makes replicability of these studies problematic and leaves little space for alternative interpretations or methodological approaches.

Hacker efforts to redecentralize computational complexity in lifeworld Internet also inherently forecast new methodological challenges: whereas the use of personal data for research driven by commercial purposes raises ethical questions (e.g. Lanier 2014), personal data about user lifeworlds as managed within decentralized alternative rationalizations is by design owned and guarded by each individual user, making access to it for research purposes problematic. As a thought experiment, if at some point in the near future every Internet user would be using exclusively decentralized Internet affordances, the same opaqueness to large-scale surveillance that is often mentioned in hacker discourses as a desirable property of decentralized architectures would deprive legitimate research efforts from any useful data beyond what could be gathered on a case-by-case basis through personal interviews. Recognizing this, some hacker projects are starting to develop voluntary, user-controlled data reporting and gathering issues (e.g. the indie-stats project[126]), while methods are being developed within the field of statistics and data science to allow meaningful analysis of large data sets while preserving privacy, through methods related to the concept of 'differential privacy'[127] (Dwork 2006; Dwork and Naor 2008; Dwork and Roth 2014); the issue itself and the proposed voluntary approaches to anonymous/pseudonymous data disclosure, however, also make very visible the power imbalances implicit in the proprietarization of user data by private corporations: this is often performed through legal agreements that force users to surrender ownership of and control over their own data, making it available exclusively for proprietary purposes.

The substantial opaqueness of individual lifeworlds that Internet research focused on everyday life unavoidably faces may contribute to making the domain of the basic, mun-

[126] `https://github.com/bear/indie-stats`

[127] Dwork and Roth 2014 provides a thorough overview of methods developed since 2006 in this field. This work also includes an accessible definition of differential privacy: *"Differential privacy" describes a promise, made by a data holder, or curator, to a data subject: "You will not be affected, adversely or otherwise, by allowing your data to be used in any study or analysis, no matter what other studies, data sets, or information sources, are available." At their best, differentially private database mechanisms can make confidential data widely available for accurate data analysis, without resorting to data clean rooms, data usage agreements, data protection plans, or restricted views. [...] Differential privacy addresses the paradox of learning nothing about an individual while learning useful information about a population* (ibid., p5).

dane, unglamorous 'stuff' that users deal with in everyday life on the Internet, and the research focus on this domain, seem as 'first world problems', when compared with the spectacular visibility of Internet uses through large political struggles, human rights activism, education, health and other public and political domains: channeling efforts towards developing computational agency that could help to empower politically and economically subaltern groups may reasonably be considered a better focus than developing decentralized alternatives to systems that—despite their obvious issues—reasonably succeed in addressing everyday user needs. On the contrary, my suggestion is that lack of control over software in the personal sphere is systemic in 'first world contexts', and through this lack of control computational agency is often implicitly traded for convenience and assigned to advertising companies that are a vital part of the global capitalist systems of exploitation of natural resources, human dignity and personal freedom. Nevertheless, lifeworld Internet is not necessarily limited to the relatively privileged practices observed through my fieldwork; for example, as the recent visibility of the Syrian humanitarian crisis has brought to public attention, the Internet and mobile devices seem to be playing an increasingly important role in the distressful life stories of refugees: the Techfugees project[128] has been gathering developers interested in contributing expertise where required to address pressing everyday needs of refugees in search of information and trying to keep in touch with displaced relatives. Although such initiatives may risk to develop 'solutionist' approaches, research focused on the very essential use of Internet technologies throughout humanitarian crises would constitute further meaningful exploration of lifeworld Internet: not by focusing on technical support to *responses* (such as the better known and more established OpenStreetMap hackathons aimed at mapping in detailed areas where emergency response is needed) but on how *individual* needs can be better understood and addressed with the hope of informing, through ethnographic research, interventions aimed at providing support to individuals in their daily struggles.

I conclude these final remarks by reconnecting to Allen's (2013) analysis of the introduction of discourses of versions through the Web 2.0 narrative I referenced in Chapter 4: his own suggestions for further research sensibly highlight how the irreducible unique-

[128]`http://www.techfugees.com/`.

ness of each individual's lifeworld should be captured through individual stories, which together compose a distributed account of 'socio-technological engagement':

> Rather than conceiving of internet history as an external process of change in media, technology and communication, [...] a fruitful new direction for internet and media researchers is to discover how individuals found a place in their lives *for* the changes that technology brings. As a result, we would know more of individual users' agency in their own historicity, understanding how their own self-histories interweave with the history of technology. The methods and perspectives afforded by oral history would be a very effective way of conducting this research [...]. In the end such research would enable us to understand the internet not as something that demands its own history, but as distributed, multiple fragments of socio-technological engagement, memorialized in and deeply significant to the lives of the people using it. (Allen 2013, p271)

Appendix A

Profiles of students interviewed during fieldwork (2010-2011)

Min-hee MA student (Fine arts). Female. Age range: 25-34.

Min-hee came to study in London from South Korea. She is used to always on, high–speed home and mobile Internet connections in Korea, and uses a smartphone in London to keep in touch with family and friends back home. She strives to keep a tight control over her social time online as she hates being interrupted via instant messaging on social networks, which she sees as very popular amongst her peers.

Kathy BA student (Politics and History). Female. Age range: <25.

Kathy got interested in computers relatively recently, in order to keep in touch with friends she won't be seeing regularly now that she moved to London for her studies. She started blogging for family and friends during her gap year, when she volunteered in a village school in Kenya, posting updates from an Internet café during weekend breaks at the nearby beach.

Ian PGCE student (Education). Male. Age range: <25.

Ian grew up in the local neighbourhood and is not a keen Internet user: he only reads local blogs and is not active on social networks. He doesn't find the Internet of much use at this stage of his life.

Hye-jung MA student (Cultural studies). Female. Age range: 35-44.

Hye-jung is a travel and food writer and photographer from Singapore. She has been keeping a blog since the early 2000s, which she updates regularly with random thoughts, reviews of books and films, rough cut versions of her travel and food writing.

Christina MA student (Sociology). Female. Age range: <25.

Christina is an avid Facebook user; she is constantly updating her public profile, meticulously curating what others can see about her (photos, text, preferences). She is interested in the link between self and technologies academically: this is the focus of her MA course options.

Inaki MA student (Sociology). Male. Age range: 25-34.

Inaki worked for a few years in his town in the Basque Country in order to save for his studies in London. He set up some private Internet infrastructure to be able to talk to friends and family through his own VoIP system, avoiding Skype and other mainstream services, as he is concerned about their opaque handling of user data and content. Learning to set up simple Internet infrastructure got him interested in 'hacking', and he keeps learning and adding little bits to his private infrastructure so that he can avoid relying on proprietary services in his everyday life.

Peter MA student (Design and critical practice). Male. Age range: 25-34.

Peter has done some occasional web development work and is interested academically in designing software experiences that can help users to reflect about their 'here and now'. He is a keen user of free software and he supports the politics of free software. He cycled from England to India with a friend recently and that has made him understand how little he can live with on a daily basis, focusing instead on spiritual values.

Rosa MA student (Anthropology). Female. Age range: 25-34.

Rosa was working as a documentary film–maker in Barcelona before taking a study break in London. She enjoys the ability to stay in touch with friends through social networks, but is not constantly online: she sees SNSes as an useful augmentation of her 'offline sociality'. She is keen to learn how to reconfigure her Internet environment and to understand how using various sites, apps and services can complement her daily tasks.

Adlina MA student (Design and critical practice). Female. Age range: 25-34.

Adlina is an avid Internet user, interested in data-driven journalism and data visualizations. She has been learning how to use data visualization tools and has some experience with software coding, in order to process and analyze data for her projects.

Rachel MA student (Music). Female. Age range: 25-34.

Rachel gave up her job as musician at a music label, which she saw as too restrictive for her creativity. She actively questions the apparent inability of the music industry to embrace Internet technologies, and she is experimenting with strategies, tools and apps that allow her to establish a direct link with the fans of her creative work.

Tawfiq BA student (Mathematics and computing). Male. Age range: <25.

Tawfiq is a keen Internet user. He doesn't own a smartphone as he cannot afford one, so he tries to get connected to wireless network wherever he is, in order to be always available on social networks, through which he engages in constant conversations with friends. He learnt how to use computers and how to code largely by himself until now; although he thinks that GNU/Linux would allow him to have closer control over reconfiguring his Internet environment, he uses Windows as his daily operating system, finding it more convenient: he is happy with tweaking what he can through a visual interface rather than by writing code.

Emir PhD student (Media and communications). Male. Age range: 25-34.

Emir used to work in radio broadcasting before taking a study break for his PhD. He researches advertising in public spaces. He is currently limiting his Internet activity due to lack of time: he mainly follows updates from his friends about the political situation in his home country but he doesn't engage in the related conversations on social networks.

Maysam BA student (Anthropology). Female. Age range: <25.

Maysam is a keen user of social networks: she splits her presence between Orkut, where she keeps in touch with people she became friends with while travelling in Brazil, and Facebook—for everything else. She exchanges information, notes and questions with coursemates via Facebook groups, preferring this to the university's own learning platform.

Markus PhD student (Media and communications). Male. Age range: 25-34.

Markus is an Internet hacktivist, media artist and PhD student. His research project is a participatory documentary film–making programme in the local community, through which he runs workshops teaching people how to shoot, edit and publish video. He is deeply concerned about power issues involved in the structure of mainstream Internet; with the help of friends (he is not an advanced coder or systems administrator himself) he is progressively setting up independent Internet infrastructures both for his work/art/research projects and for his private life.

Appendix B

Sources for the textual analysis of hacker publics and their projects

A brief overview of the sources used for the analysis developed in Chapters 7 and 8 was included in the respective Introductions, with the aim of outlining the textual[129] environments through which my third stage of fieldwork was conducted, and to highlight the distributed nature (in terms of geography and cultural diversity of the hacker contexts explored) of the source materials.

A fuller overview of my sources is provided in this appendix, grouping sources by type and including, for each group, a brief description and a discussion of its significance within the overall research focus. The aim of this appendix is to provide additional context and clarity to the choice of sources, to highlight their scope and diversity (both between and within the different groups) and ultimately to allow the reader to understand how I approached the network of actors and texts through which the computational turn of the Internet, recursive publics of hackers and alternative rationalizations of Internet infras-

[129] In a broad sense: including video and audio recordings as well as software source code, alongside a limited number of observations and discussions conducted in person at hacker conventions.

tructure were followed and analyzed.

As outlined in Chapter 3 (Research methods), I did not employ a statistical sampling of sources and I did not aim to perform a very large–scale textual analysis over an extensive textual corpus. On one hand, this would have constituted a monumental effort well beyond the scope of the present work due to the extremely vast amount of conversations and other materials available and created every day. The availability of growing amounts of relevant materials as video or audio recordings only, moreover, would have posed the additional challenge of dealing with only a subset of the relevant sources delimited by technical limitations rather than through substantive research decisions: only rarely freely accessible textual transcripts were available, and although computational automated close captioning is increasingly more accurate, I found this to still be unsuitable, in practice, for the purposes of analyzing large amounts of text reliably.

More importantly, however, I considered such kind of large–scale textual analysis to not be adequate for my aim of tracing the development of shared systems of meanings and values throughout the establishment and reproduction of the recursive public of hackers: besides the analysis of the *contents* of hacker discourses, my methods needed to allow me to focus on following the actors themselves, as well as their attention, in order to try to trace:

- how topics of shared interest are identified, shared and developed, across space, time and contexts
- the reputation of hackers who write or speak about topics relevant to my enquiry
- how the multiple channels through which discourse is articulated contribute, each in specific ways, to the development of shared representations (therefore including the role of the materiality of the media involved, especially when these include software source code)

The intent, therefore, was not to pursue an 'objective' and articulated portrayal of technologies and strategies, but to analyze the recursive public's construction and discussion of these technologies and strategies, in connection to motivations and agendas, and against the backdrop of hegemonic technical form and discourse.

Whereas the high level of technical detail in Chapters 7 and 8 may lead the reader to question the role of the articulation of alternative rationalizations beyond that of a survey of recent projects, the following outline of source materials should hopefully clarify that these chapters' detailed narratives actually constitute curated renderings of shared representations coalesced around motivations and attention of redecentralization hackers, assembled into manageable units from a multiplicity of sources.

Finally, although these sources are here grouped logically by type, it is important to highlight that I often followed actors and discourses across these groups: for example, an interview in a printed magazine may have led me to review the interviewee's blog or Twitter stream in order to try to identify further relevant materials.

B.1 Magazines (printed or digital)

This group of sources is composed of a growing array of magazines (printed or digital) that aim to step back from the technical details of software engineering, web development, design and management of computational infrastructure to focus instead on the 'people behind bits and pixels'[130] and on current ideas, discussions and focus of attention in hacker circles.

Amongst the magazines consulted: Offscreen Magazine (printed); The manual (printed and digital); The pastry box project (digital). A range of other magazines was consulted typically to read specific articles: for example, articles cited elsewhere (blogs, social news sites).

Significance These sources highlight the increasing attention to non–technical materiality amongst hackers and constituted a turning point in my understanding of the ongoing process of establishment and reproduction of a visible hacker public intent in identifying the social value of its software development work.

Moreover, as part of the recursive public of hackers myself, these magazines also contributed to developing my own political awareness of the contestations around Internet infrastructure; a trace of my autoethnographic involvement (cfr. section on

[130] This is the tagline of one such publications, Offscreen Magazine.

autoethnography in Chapter 3) is my cameo appearance in the blog of the editor of Offscreen Magazine, publicly exposing through my writing of a letter to the editor my relationship with the discourses developed in the magazine (rota 2015).

B.2 Social news websites

This group of sources is composed of a small set of *social news* websites collectively co–curated by their own readers (often with the aid of internal or external moderators).

The two main sources consulted were Hacker News and web– and Internet–focused 'subreddits' on Reddit; other social news websites were occasionally consulted, mainly whilst following discussions that had originated elsewhere. Although the use of just two main sources within this group may lead to questioning the possibility that they may contribute to developing a sort of discursive 'filter bubble' (Pariser 2011), this choice results from a narrowing down over the timespan of my fieldwork of an initially broader range of social news sites, as I observed that most of the discussions relevant to my research focus were being developed on these two sites. As outlined in chapter 7, moreover, a degree of (permeable) closure of the universe of discourse is nevertheless characteristic of recursive publics, and as such fundamentally unavoidable: even a cursory glance of the discussion threads attached to the most popular topics, however, clearly highlights how profound disagreements nevertheless regularly unfold even on these sites.

The use of another popular site, Slashdot, proved problematic as the quality and relevance of its discussions oscillated throughout the timespan of my fieldwork (especially after the resignation of its founder and editor in 2011 and a 2012 change in ownership), and because of its somewhat narrower focus on science and technology rather than on the consistently broader ranges of domains discussed on Reddit and Hacker News.

Significance Differently from discussion–oriented forums (or even recent question–and–answer forums such as those analyzed in Chapter 7), these sites are focused on the sharing, upvoting/downvoting and commenting of external links: in other words, the content being discovered and shared as relevant, and whose relevance to the group is signalled and discussed through voting and comments, does not originate on the site itself (Graham 2015). By improving discoverability of arbitrary content deemed relevant to the group and by accommodating group discussion of relevance, social news sites constitute interesting (social) sites of translation between hacker publics and broader social, technical and political concerns. Software projects considered worthy of attention are often discussed on these sites, frequently leading me to sources within the 'software code' group of source materials outlined below. Furthermore, the sense of shared relevance of Hacker News, specifically, amongst hackers, is also signalled by the occasional use of Hacker News itself as the place where individual hackers direct discussion about their own blog posts: rather than hosting a commenting system on the blog itself, some popular blogs leverage the community dynamics of Hacker News and simply display a 'Discuss on Hacker News' button at the bottom of each blog post.

B.3 Blogs and microblogs of individual hackers

This group of sources includes the personal blogs of hackers, makers, developers, designers, as well as posts on microblogging sites (e.g. Twitter or decentralized alternatives). Posts by Internet 'pundits' were also consulted, when referenced as relevant from other sources, although not systematically unless these pundits were also clearly part of the recursive hacker public, as their insights may be useful for the understanding of specific phenomena but the authors would miss the core requirement, within my research focus, of being situated within the public being investigated. This group of sources obviously includes an extremely heterogeneous and broad array of blogs. I did not survey them all systematically and I did not (except for an initial period) use computational aids such as feed readers to manage this informational flow: on one hand, given the number of blogs and posts involved, and their spread over global timezones, the amount of information was simply not manageable, even considering the significant share of posts not relevant in any way to my focus; on the other hand, I felt that only by carefully following recommendations, cross citations and other signals of relevance I could effectively understand the construction of shared representations. I did, however, obviously return to blogs even without the constant computational 'prodding' of a feed reader: each new visit was akin to a weak personal reconnection, quickly 'catching up' on ideas and conversations that each hacker may have been involved in recently.

Significance This group of sources was ostensibly the most complicated to manage because of its heterogeneity; however this same factor also allowed me to follow individual hackers in their own personal communication spaces, where they would usually feel free from the explicit and unspoken rules of curated aggregators and platforms. Sources from this group generally provided the most relevant insights through my fieldwork: whereas other types of sources (social news sites, conference presentations) often constituted a point of entry and discovery of the work and writings of individual hackers, I often gained a better understanding of their motivations and values by following their writing on their personal blogs.

B.4 Engineering blogs

This group of sources is constituted by group blogs to which technical staff of Internet companies, startups or Internet–focused departments within larger organizations[131] contribute. Topics discussed are often strictly technical (focusing on the company's ways of solving design, performance, security and infrastructure challenges), although it is often possible to glean interesting information on the shared values and on the culture of the company.

Similarly to the previous group, even considering only the few hundreds engineering blogs widely known[132], following them all systematically would have been practically impossible, especially considering the large amount of content not relevant to my research: I often read and filed individual posts referenced from discussions elsewhere, and I occasionally returned to specific blogs, for example when expecting a company's statement on some discussion unfolding throughout hacker discourses at a given time.

[131] For example, the web operations group of a newspaper such as The Guardian.

[132] GitHub user *kilimchoi* started maintaining in 2013 a list of such blogs in a GitHub repository open to contributions (`https://github.com/kilimchoi/engineering-blogs`); I learnt about this list through a post on Hacker News. The list includes just over 450 blogs at the time of writing.

Significance Besides useful insights into values and culture of specific companies, engineering blogs were typically consulted to improve my understanding of new technical strategies being discussed elsewhere, where these had originated at a specific company.

B.5 Conferences and hacker conventions

This group of sources includes video or audio recordings of presentations given by hackers at technical conferences and hacker conventions. Whenever transcripts were available, I usually referred to these as quicker to parse, although I sometimes nevertheless watched parts of presentations whose content I had read, in order to try to gain a better understanding of the speaker's intent through their non–verbal cues.

Besides a few key conferences or conference series whose programmes I browsed systematically (for example WebStock, DebConf, FOSDEM, IndieWebCamps, Chaos Communication Congress), I usually accessed presentations by searching for specific topics or through recommendations found in discussions elsewhere or through the event archives on the Huffduffer audio aggregator.

As in other groups of sources, I focused on presentations that explored the social significance of projects and practices, although more technical presentations occasionally helped me to better understand the detailed materiality of strategies whose computational structure was not familiar to me.

Significance Due to the increasing visibility of presentations at conferences—as more of these are recorded for future reference and recommendations to watch specific presentations can therefore be easily spread through social media—this group of sources was particularly interesting for my research: besides their informational content and their role as introducers to the work and motivations of hackers previously unknown to me, they often provided a visible indication of the popularity of topics and speakers (through the view counters of each video, where available) and a connection to the associated discussions (through comment threads by the side of the videos themselves or on social news sites) and to the related processes of sense making and scrutiny being operated by the public of watchers.

Moreover, routine recording of sessions at conferences and conventions not only makes available to a global public discussions that happened in person in specific locations, but also makes speakers and participants ultimately more accountable for the content and formal quality of their contributions: whereas unfortunate remarks or controversial statements may have remained circumscribed to the audience present at events not being recorded a few years ago, any session that is recorded and archived will instead provide evidence of controversies as long as it remains archived for watching: further research will be needed to validate the social relevance of this, but several hackers who were part of my fieldwork discussed how they felt that the growing corpus of recorded presentations and discussions was contributing, in their view, to the constitution of a shared hacker discourse.

B.6 Interviews and podcasts

This group of sources includes interviews to individual hackers or recorded conversations with small groups of hackers (in the latter case, typically within the format of podcasts).

A core set of interviews used was the one curated by the Redecentralize project[133], focusing on individual hackers leading, or involved in, redecentralization projects. Other interviews and group conversations in podcasts were consulted occasionally, mainly as a way for me to get to know directly from developers (rather than through the explanations of external commentators) new hacker projects.

Significance The Redecentralize project, through its curated set of interviews and through the discussion list associated to the project, constituted a key source of information, thanks to its role in bringing together redecentralization hackers and interested users within a single site. The specific focus of their set of interviews—discussing with key hackers the motivations behind their projects and how they felt their social value is, often with explicit reference to the domain of users' everyday life—was highly relevant to the topics of Chapter 8. Other interviews from other sources

[133] `http://redecentralize.org/`

often helped me to get acquainted with new projects and with their authors, constituting a starting point for further exploration through other, connected sources (for example, by browsing a project's website or source code repositories).

B.7 Source code and project websites

This group of sources includes source code repositories and project websites of projects which were developing alternatives to mainstream Internet.

Significance I browsed a large quantity of project websites in order to understand their technical form as well as their relevance to lifeworld Internet and as alternatives to mainstream Internet. Reading software source code proved to be a central and precious part of my broad textual analysis; I did not employ quantitative or computational methods for the analysis of code (although I had experience of this through my work on rota and Pozzi 2013), as my main aim was not to understand the technical details of software code, but rather the ways in which the computational turn is reshaping the way code is written, documented, distributed and made available for use: often the most interesting parts of a code repository proved to be the files listing a project's dependencies on external FLOSS libraries and the configuration files for various scripts and systems used to computationally manage the code (tests, installation, etc.). The growing attention to user documentation and to providing simple ways for casual users to install and evaluate the usefulness of FLOSS projects was a key learning experience in itself, which helped me to conceptualize and develop the sections on user engagement, convenience and trust, and computational capacity of Chapter 8.

B.8 Academic and professional journals

This group of sources includes articles in academic and professional journals. In most cases these were computer science articles (and a few doctoral dissertations), and in some cases more narrative articles (mostly on professional, rather than academic, journals) about the contexts within which projects were being developed.

Significance I used these sources mainly to gain a technical understanding of material details of relevant software projects, to better inform the exploration of hacker motivations and goals which I performed through other types of sources.

References

Abbate, Janet (1999). *Inventing the Internet*. Cambridge, Mass: MIT Press.

Allan, Stuart (2006). *Online news: journalism and the Internet*. Open University Press.

Allen, Matthew (2013). 'What was Web 2.0? Versions as the dominant mode of internet history'. In: *New Media & Society* 15.2, pp. 260–275.

Alleyne, Brian (2011). '"We are all hackers now": critical sociological reflections on the hacking phenomenon'. In: *Under Review*, pp. 1–32. URL: http://eprints.gold.ac.uk/6305 (visited on June 30, 2015).

Anderson, Chris (2004). 'The Long Tail'. In: *Wired* 12.10. URL: http://archive.wired.com/wired/archive/12.10/tail.html (visited on Dec. 7, 2014).

Anderson, L. (2006). 'Analytic Autoethnography'. In: *Journal of Contemporary Ethnography* 35.4, pp. 373–395.

Andersson Schwarz, Jonas (2013). *Online File Sharing: Innovations in Media Consumption*. New York: Routledge.

Andreessen, Marc (2011). 'Why Software Is Eating The World'. In: *Wall Street Journal*. URL: http://on.wsj.com/1w2FbVs (visited on Oct. 4, 2015).

Andy (2013). *Kim Dotcom Releases New Raid Footage Captured By In-House CCTV*. TorrentFreak. URL: https://torrentfreak.com/kim-dotcom-releases-new-raid-footage-captured-by-in-house-cctv-130613/ (visited on Jan. 13, 2015).

Atkinson, Paul (2006). 'Rescuing Autoethnography'. In: *Journal of Contemporary Ethnography* 35.4, pp. 400–404.

Atkinson, Robert (1998). *The life story interview*. Thousand Oaks, CA: Sage Publications.

Atwood, Jeff (2008). *Introducing Stackoverflow.com*. Coding Horror. URL: http://blog.codinghorror.com/introducing-stackoverflow-com/ (visited on Aug. 3, 2015).

Aurigi, Alessandro (2005). *Making the digital city: the early shaping of urban Internet space*. Ashgate.

Back, M. D., J. M. Stopfer, et al. (2010). 'Facebook Profiles Reflect Actual Personality, Not Self-Idealization'. In: *Psychological Science* 21.3, pp. 372–374.

Bakardjieva, Maria (2005). *Internet Society. The internet in everyday life*. London: Sage.

— (2012). 'Subactivism: Lifeworld and Politics in the Age of the Internet'. In: *(Re)Inventing The Internet: Critical Case Studies*. Ed. by Andrew Feenberg and Norm Friesen. Sense Publishers, pp. 85–108.

Bakardjieva, Maria and Georgia Gaden (2012). 'Web 2.0 Technologies of the Self'. In: *Philosophy & Technology* 25.3, pp. 399–413.

Balkan, Aral (2014). 'Brave New World — Ind.ie Summit keynote'. Ind.ie Summit. Brighton, UK. URL: https://ind.ie/archive/summit/videos/opening-keynote/index.html (visited on Nov. 26, 2015).

— (2015). 'Privacy, Data, Democracy'. The Conference. Malmö. URL: http://videos.theconference.se/aral-balkan-privacy-data-democracy (visited on Nov. 18, 2015).

Barrett, Ryan (2014). *Indie stats*. snarfed.org. URL: https://snarfed.org/indie-stats (visited on Sept. 9, 2015).

Baym, Nancy K. (1995). 'From practice to culture on Usenet'. In: *The cultures of computing*. Ed. by Susan Leigh Star. Oxford: Blackwell.

— (1998). 'The Emergence of On-Line Community'. In: *CyberSociety 2.0: Revisiting Computer-Mediated Community and Technology*. Ed. by Steven G. Jones. New Media Cultures. Thousand Oaks, CA: Sage Publications.

Bearman, Joshuah (2015a). 'The Untold Story of Silk Road, Part 1'. In: *WIRED*. URL: http://www.wired.com/2015/04/silk-road-1/ (visited on Oct. 4, 2015).

— (2015b). 'The Untold Story of Silk Road, Part 2: The Fall'. In: *WIRED*. URL: http://www.wired.com/2015/05/silk-road-2/ (visited on Oct. 4, 2015).

Bell, David and Barbara M. Kennedy, eds. (2000). *The cybercultures reader*. London: Routledge.

Belleflamme, Paul, Thomas Lambert, et al. (2014). 'Crowdfunding: Tapping the right crowd'. In: *Journal of Business Venturing* 29.5, pp. 585–609.

Berkholz, Donnie (2013). *The size of open-source communities and its impact upon activity, licensing, and hosting*. Donnie Berkholz's Story of Data. URL: `http://redmonk.com/dberkholz/2013/04/22/the-size-of-open-source-communities-and-its-impact-upon-activity-licensing-and-hosting/` (visited on Dec. 4, 2014).

Berners-Lee, Tim (1998). *The World Wide Web: A very short personal history*. URL: `http://www.w3.org/People/Berners-Lee/ShortHistory.html` (visited on Apr. 21, 2015).

Berry, David M. (2004). 'The contestation of code: A preliminary investigation into the discourse of the free/libre and open source movements'. In: *Critical Discourse Studies* 1.1, pp. 65–89.

— (2011). 'The Computational Turn: Thinking About the Digital Humanities'. In: *Culture Machine* 12.

— (2014). *Critical Theory and the Digital*. New York; London: Bloomsbury Academic.

Bijker, Wiebe E, Thomas Parke Hughes, et al., eds. (1987). *The Social construction of technological systems: new directions in the sociology and history of technology*. Cambridge, Mass: MIT Press.

blue_beetle (2010). *If you are not paying for it, you're not the customer; you're the product being sold.* metafilter. URL: `http://www.metafilter.com/95152/Userdriven-discontent` (visited on Nov. 18, 2015).

Boellstorff, Tom (2008). *Coming of age in second life: an anthropologist explores the virtually human*. Princeton, NJ: Princeton University Press.

Boellstorff, Tom, Bonnie Nardi, et al. (2012). *Ethnography and Virtual Worlds: A Handbook of Method*. Princeton University Press.

Bolt Beranek and Newman Inc (1981). *A history of the ARPANET: The First Decade*. URL: `http://www.dtic.mil/cgi-bin/GetTRDoc?AD=ADA115440` (visited on Apr. 12, 2015).

Bolter, J. and Richard Grusin (1999). *Remediation: understanding new media*. Cambridge, Mass: MIT Press.

Bolychevsky, Irina (2013a). *Interview #5: Jacob Cook - arkOS*. In collab. with Jacob Cook. URL: `http://redecentralize.org/interviews/2013/09/13/05-jacob-arkos.html` (visited on Nov. 26, 2015).

— (2013b). *Interview #6: Christopher Webber - Media Goblin*. In collab. with Chris Webber. URL: `http://redecentralize.org/interviews/2013/10/13/06-chris-mediagoblin.html` (visited on Nov. 25, 2015).

Bourdieu, Pierre (1984). *Distinction: a social critique of the judgement of taste*. London: Routledge & Kegan Paul.

boyd, danah m. (2008). 'Taken out of context'. Doctoral dissertation. University of California, Berkeley.

— (2014). *It's Complicated: The Social Lives of Networked Teens*. Yale University Press.

boyd, danah m. and Nicole B. Ellison (2007). 'Social Network Sites: Definition, History, and Scholarship'. In: *Journal of Computer-Mediated Communication* 13.1, pp. 210–230.

Brach, Kai (2014). *The Unbalanced Line-Up*. Offscreen Magazine Blog. URL: `http://blog.offscreenmag.com/post/73171666876/the-unbalanced-line-up` (visited on Jan. 25, 2015).

Bradbury, Danny (2013). 'The problem with Bitcoin'. In: *Computer Fraud & Security* 2013.11, pp. 5–8.

Brannen, Julia (1995). 'Combining qualitative and quantitative approaches: an overview'. In: *Mixing Methods: Qualitative and Quantitative Research*. Ed. by Julia Brannen. Ashgate.

Bromberg, Heather (1996). 'Are MUDs Communities? Identity, Belonging and Consciousness in Virtual Worlds'. In: *Cultures of Internet: Virtual spaces, real histories, living bodies*. Ed. by Rob Shields. London: Sage.

Brooks, F. P. (1987). 'No silver bullet'. In: *Computer* 20.4, pp. 10–19.

Butler, Judith (1993). *Bodies that matter: on the discursive limits of "sex"*. New York: Routledge.

Callon, Michel (1984). 'Some elements of a sociology of translation: domestication of the scallops and the fishermen of St Brieuc Bay'. In: *The Sociological Review* 32.1, pp. 196–223.

Carr, Nicholas (2008). 'Is Google Making Us Stupid?' In: *The Atlantic*. URL: `http://www.theatlantic.com/magazine/archive/2008/07/is-google-making-us-stupid/306868/` (visited on Dec. 8, 2014).

— (2011). *The Shallows: What the Internet Is Doing to Our Brains*. W. W. Norton & Company.

Case, Amber (2015). *Why We All Need to Make the Internet Fun Again*. Re/code. URL: `https://recode.net/2015/07/17/why-we-all-need-to-make-the-internet-fun-again/` (visited on July 25, 2015).

Castells, Manuel (2012). *Networks of outrage and hope : social movements in the Internet age*. Cambridge, UK ; Malden, MA: Polity.

Çelik, Tantek (2013). *The Homebrew Computer Club 38th Reunion: Inspiration To Try A Variant*. URL: `http://tantek.com/2013/322/b1/homebrew-computer-club-reunion-inspiration` (visited on July 21, 2015).

— (2015). *IndieWebCamp - Founders*. IndieWebCamp. URL: `http://indiewebcamp.com/founders` (visited on June 13, 2015).

Cerf, Vinton G. and Robert E. Kahn (1974). 'A Protocol for Packet Network Intercommunication'. In: *IEEE Transactions on Communications* 22.5, pp. 637–648.

Certeau, Michel de (1984). *The practice of everyday life*. Berkeley: University of California Press.

Coglan, James (2013). *Why GitHub is not your CV*. The If Works. URL: `https://blog.jcoglan.com/2013/11/15/why-github-is-not-your-cv/` (visited on July 28, 2015).

Coleman, E. Gabriella (2012a). *Coding Freedom: The Ethics and Aesthetics of Hacking*. Princeton University Press.

— (2012b). 'Phreaks, Hackers, and Trolls: The Politics of Transgression and Spectacle'. In: *The social media reader*. Ed. by Michael Mandiberg. NYU Press, pp. 99–119.

Coleman, E. Gabriella (2014). *Hacker*. In: *The Johns Hopkins Encyclopedia of Digital Textuality*. URL: `http://gabriellacoleman.org/wp-content/uploads/2013/04/Coleman-Hacker-John-Hopkins-2013-Final.pdf` (visited on Nov. 16, 2015).

Combrinck, Tanya (2012). *Remy Sharp on learning by breaking stuff*. In collab. with Remy Sharp. URL: `http://www.creativebloq.com/html5/remy-sharp-learning-breaking-stuff-11125970` (visited on Oct. 25, 2015).

Corbet, Jonathan (2015). 'Development statistics for the 4.2 kernel'. In: *Linux Weekly News*. URL: `https://lwn.net/Articles/654633/` (visited on Aug. 19, 2015).

Coupland, Douglas (2007). *JPod*. London: Bloomsbury.

Crain, Matthew (2014). 'Financial markets and online advertising: reevaluating the dotcom investment bubble'. In: *Information, Communication & Society* 17.3, pp. 371–384.

Curran, James (2012). 'Reinterpreting the internet'. In: *Misunderstanding the internet*. Ed. by James Curran, Natalie Fenton, et al. Routledge.

Custer, C. (2015). 'Mark Zuckerberg is trying to hoodwink India'. In: *Tech in Asia*. URL: `https://www.techinasia.com/disconnect-mark-zuckerberg-scam-indias-poor` (visited on Jan. 4, 2016).

Dabbish, Laura, Colleen Stuart, et al. (2012). 'Social coding in GitHub: transparency and collaboration in an open software repository'. In: *Proceedings of the ACM 2012 conference on Computer Supported Cooperative Work*. ACM, pp. 1277–1286.

Dahlgren, Peter (2013). *The political web : media, participation and alternative democracy*. Houndmills, Basingstoke, Hampshire: Palgrave Macmillan.

Debois, Patrick (2009). *Charting out devops ideas*. URL: `http://www.jedi.be/blog/2009/12/22/charting-out-devops-ideas/` (visited on Jan. 13, 2016).

Deering, Stephen E. and Robert M. Hinden (1998). *RFC2460: Internet Protocol, Version 6 (IPv6) Specification*.

Denzin, Norman (2006). 'Analytic Autoethnography, or Deja Vu all Over Again'. In: *Journal of Contemporary Ethnography* 35.4, pp. 419–428.

Denzin, Norman K. (2014). *Interpretive Autoethnography*. SAGE Publications.

DiNucci, Darcy (1999). 'Fragmented future'. In: *Print* 53.4, pp. 32, 221–222. URL: `http://darcyd.com/fragmented_future.pdf` (visited on Mar. 17, 2015).

Discourse project (2013). *Discourse - About*. URL: `https://web.archive.org/web/20130306124312/http://www.discourse.org/about/` (visited on Mar. 6, 2013).

drbob07 (2012). *Ideals behind Retroshare*. RetroShare Blog. URL: `https://retroshareteam.wordpress.com/2012/10/26/ideals-behind-retroshare/` (visited on Nov. 24, 2015).

Duloz, Alex (2015). *In Conclusion*. The Pastry Box Project. URL: `https://the-pastry-box-project.net/alex-duloz/2015-december-31` (visited on Jan. 6, 2016).

Dwork, Cynthia (2006). 'Differential privacy'. In: *Proceedings of the International Col- loquium on Automata, Languages and Programming*. ICALP, pp. 1–12.

Dwork, Cynthia and Moni Naor (2008). 'On the difficulties of disclosure prevention in statistical databases or the case for differential privacy'. In: *Journal of Privacy and Confidentiality* 2.1, pp. 93–107.

Dwork, Cynthia and Aaron Roth (2014). *The Algorithmic Foundations of Differential Privacy*. Vol. 9. Foundations and Trends in Theoretical Computer Science 3-4. now publishers.

Elmer-DeWitt, Philip and David S. Jackson (1993). 'First nation in cyberspace'. In: *Time* 142.24, p. 62.

Ensmenger, Nathan (2010). *The computer boys take over: computers, programmers, and the politics of technical expertise*. Cambridge, Mass: MIT Press.

Facebook (2015). *Facebook Reports Third Quarter 2015 Results - Facebook*. URL: `http://investor.fb.com/releasedetail.cfm?ReleaseID=940609` (visited on Nov. 26, 2015).

Fahl, Sascha, Marian Harbach, et al. (2012). 'Helping Johnny 2.0 to encrypt his Facebook conversations'. In: *Proceedings of the Eighth Symposium on Usable Privacy and Security*. ACM.

Feenberg, Andrew (1995). *Alternative Modernity: The Technical Turn in Philosophy and Social Theory*. Berkeley: University of California Press.

— (1999). *Questioning technology*. London: Routledge.

— (2002). *Transforming technology: a critical theory revisited*. Oxford University Press.

Feenberg, Andrew (2012). 'Introduction: Toward a critical theory of the Internet'. In: *(Re)Inventing The Internet: Critical Case Studies*. Ed. by Andrew Feenberg and Norm Friesen. Sense Publishers, pp. 3–18.

Feenberg, Andrew and Norm Friesen, eds. (2012). *(Re)Inventing the Internet: Critical Case Studies*. Sense Publishers.

Felten, Ed (2006). *Taking Stevens Seriously*. Freedom to Tinker. URL: `https://freedom-to-tinker.com/blog/felten/taking-stevens-seriously/` (visited on Apr. 23, 2015).

Flickenger, Rob (2003). *Building Wireless Community Networks*. O'Reilly Media.

Floridi, Luciano (2014). *The Fourth Revolution: How the Infosphere is Reshaping Human Reality*. Oxford University Press.

Fontana, Richard (2013). *Post open source software (POSS), licensing and GitHub*. OpenSource.com. URL: `http://opensource.com/law/13/8/github-poss-licensing` (visited on Dec. 4, 2014).

Foucault, Michel (1988). 'Technologies of the self'. In: *Technologies of the self: a seminar with Michel Foucault*. Ed. by Luther H. Martin, Huck Gutman, et al. Univ. of Massachusetts Press, pp. 16–49.

Fuchs, Christian (2012). 'Critique of the Political Economy of Web 2.0 Surveillance'. In: *Internet and Surveillance: The Challenges of Web 2.0 and Social Media*. Ed. by Christian Fuchs, Kees Boersma, et al. Routledge, pp. 31–70.

— (2014a). *Digital labour and Karl Marx*. New York: Routledge.

— (2014b). *Social media: a critical introduction*. Los Angeles: SAGE.

Garrett, Jesse James (2008). *ajax: a new approach to web applications*. Adaptive Path. URL: `https://web.archive.org/web/20080702075113/http://www.adaptivepath.com/ideas/essays/archives/000385.php` (visited on Aug. 25, 2015).

Gerbaudo, Paolo (2012). *Tweets and the streets : social media and contemporary activism*. London: Pluto Press.

GitHub (2015). *GitHub - Press*. GitHub. URL: `https://github.com/about/press` (visited on Oct. 10, 2015).

Gkotsis, George, Maria Liakata, et al. (2015). 'Leveraging Textual Features for Best Answer Prediction in Community-based Question Answering'. In: arXiv: 1506 . 02816v2. URL: http://arxiv.org/abs/1506.02816.

Governor, James (2012). *"younger devs today are about POSS - Post open source software. fuck the license and governance, just commit to github."* URL: https://twitter.com/monkchips/status/247584170967175169 (visited on Jan. 16, 2015).

Graham, Paul (2015). *Hacker News Guidelines*. URL: https://news.ycombinator.com/newsguidelines.html (visited on May 2, 2015).

Greenfield, Adam (2006). *Everyware: The Dawning Age of Ubiquitous Computing*. 1st edition. Berkeley, CA: New Riders Publishing.

— (2013). *Against the Smart City*. Do projects.

Habermas, Jürgen (1987). *The theory of communicative action. Volume 2, Lifeworld and system: a critique of functionalist reason*. Trans. by Thomas McCarthy. Cambridge, England ; Oxford, England: Polity Press.

— (2006a). 'Political Communication in Media Society: Does Democracy Still Enjoy an Epistemic Dimension? The Impact of Normative Theory on Empirical Research'. In: *Communication Theory* 16.4, pp. 411–426.

— (2006b). 'Preisrede von Jürgen Habermas anlässlich der Verleihung des Bruno-Kreisky-Preises für das politische Buch 2005'.

Hamilton, Edward and Andrew Feenberg (2012). 'Alternative rationalizations and ambivalent futures. A Critical History of Online Education.' In: *(Re)Inventing The Internet: Critical Case Studies*. Ed. by Andrew Feenberg and Norm Friesen. Sense Publishers, pp. 43–70.

Hammersley, Martyn and Paul Atkinson (2007). *Ethnography: principles in practice*. 3rd ed. London: Routledge.

Han, Sam (2011). *Web 2.0*. Routledge.

Haraway, Donna (1991). 'A Cyborg Manifesto: Science, Technology, and Socialist-Feminism in the Late Twentieth Century'. In: *Simians, Cyborgs and Women: The Reinvention of Nature*. New York: Routledge, pp. 149–81.

Heilmann, Chris (2012). *Bret Victors "Inventing on principle" – and a few things it inspired*. Mozilla Hacks – the Web developer blog. URL: https://hacks.mozilla.org/2012/04/bret-victors-inventing-on-principle-and-a-few-things-it-inspired/ (visited on Oct. 18, 2015).

Hills, Matt (2002). *Fan cultures*. London: Routledge.

Hine, Christine (2000). *Virtual ethnography*. London: SAGE.

Hippel, Eric von (2005). *Democratizing innovation*. MIT Press,

Horst, Heather A. and Daniel Miller, eds. (2012a). *Digital Anthropology*. London: Bloomsbury.

— (2012b). 'The Digital and the Human: A Prospectus for Digital Anthropology'. In: *Digital Anthropology*. Ed. by Heather A. Horst and Daniel Miller. London, New York: Berg, pp. 3–38.

Hull, Gordon (2015). 'Successful failure: what Foucault can teach us about privacy self-management in a world of Facebook and big data'. In: *Ethics and Information Technology* 17.2, pp. 89–101.

Husserl, Edmund (1970). *The crisis of European sciences and transcendental phenomenology: an introduction to phenomenological philosophy*. Evanston: Northwestern University Press.

IndieWebCamp (2014a). *monoculture*. IndieWebCamp. URL: http://indiewebcamp.com/monoculture (visited on June 14, 2015).

— (2014b). *Principles*. IndieWebCamp. URL: http://indiewebcamp.com/principles (visited on June 15, 2015).

— (2015a). *PESOS - IndieWebCamp*. URL: https://indiewebcamp.com/PESOS (visited on July 16, 2015).

— (2015b). *POSSE*. URL: https://indiewebcamp.com/POSSE (visited on July 16, 2015).

Ito, Mizuko (1997). 'Virtually embodied: the reality of fantasy in a multi-user dungeon'. In: *Internet Culture*. Ed. by David Porter. Routledge.

Jaffe, Jeff (2014). *Decision by consensus or by informed editor; which is better?* W3C Blog. URL: http://www.w3.org/blog/2014/10/decision-by-consensus-or-by-informed-editor-which-is-better/ (visited on July 26, 2015).

Jenkins, Henry (1992). *Textual poachers: television fans & participatory culture*. Studies in culture and communication. New York: Routledge.

— (2008). *Convergence Culture: Where Old and New Media Collide*. Revised edition. New York: NYU Press.

Jenkins, Henry, Sam Ford, et al. (2013). *Spreadable media: creating value and meaning in a networked culture*. NYU Press.

Jordan, Tim (2008). *Hacking: digital media and technological determinism*. Digital media and society series. Polity Press.

— (2009). 'Hacking and power: Social and technological determinism in the digital age'. In: *First Monday* 14.7.

— (2015). *Information politics: liberation and exploitation in the digital society*. Digital barricades. London: Pluto Press.

Jordan, Tim and Paul A. Taylor (2004). *Hacktivism and cyberwars: rebels with a cause?* Routledge.

Joshi, Vaidehi (2015). 'Where do you see yourself and your career in the years ahead?' In: *Offscreen Magazine* 12, p. 41.

Jungnickel, Katrina (2013). *DiY WiFi: Re-imagining Connectivity*. Palgrave Macmillan.

Kahle, Brewster (2015). *Locking the Web Open: A Call for a Distributed Web*. URL: `http://brewster.kahle.org/2015/08/11/locking-the-web-open-a-call-for-a-distributed-web-2/` (visited on Aug. 18, 2015).

Kaufman, Joshua (2002). *Zeldman on the Independent Web*. unraveled. URL: `http://unraveled.com/archives/2002/11/zeldman-on-the-independent-web` (visited on July 14, 2015).

Keen, Andrew (2006). *Jürgen Habermas on Web 2.0*. Andrew Keen. URL: `http://www.ajkeen.com/blog/2006/04/03/jurgen_habermas` (visited on Apr. 30, 2015).

— (2007). *The cult of the amateur: how today's internet is killing our culture*. Doubleday/Currency.

Keith, Jeremy (2014a). *Notes from a small website*. Adactio. URL: `https://adactio.com/journal/6826` (visited on Sept. 21, 2015).

Keith, Jeremy (2014b). 'On Indie Web'. Ind.ie Summit. Brighton, UK. URL: `https://ind.ie/summit/videos/jeremy-keith/index.html#transcript` (visited on May 9, 2015).

Kelty, Christopher M. (2008). *Two bits: the cultural significance of free software*. Durham: Duke University Press.

Kennedy, Barbara M. and David Bell, eds. (2007). *The Cybercultures Reader*. 2nd edition. Routledge.

Kitchin, Rob (2014). *Data revolution: big data, open data, data infrastructures & their consequences*. Los Angeles: Sage Publications.

Klensin, J. (2001). *RFC 2821 - Simple Mail Transfer Protocol*.

— (2008). *RFC 5321 - Simple Mail Transfer Protocol*.

Kohn, Eduardo (2013). *How forests think: toward an anthropology beyond the human*. Berkeley: University of California Press.

Kramer, A. D. I., J. E. Guillory, et al. (2014). 'Experimental evidence of massive-scale emotional contagion through social networks'. In: *Proceedings of the National Academy of Sciences* 111.24, pp. 8788–8790.

Krasner, Glenn E. and Stephen T. Pope (1988). 'A Cookbook for Using the Model-view Controller User Interface Paradigm in Smalltalk-80'. In: *Journal Object Oriented Program.* 1.3, pp. 26–49.

Kuppuswamy, Venkat and Barry L. Bayus (2014). 'Crowdfunding creative ideas: The dynamics of project backers in Kickstarter'. In: *UNC Kenan-Flagler Research Paper* 2013.

Lally, Elaine (2002). *At Home with Computers*. Bloomsbury Academic.

Lanier, Jaron (2010). *You are not a gadget: a manifesto*. Allen Lane.

— (2013). *Who owns the future?* London: Allen Lane.

— (2014). *Should Facebook Manipulate Users?* The New York Times. URL: `http://www.nytimes.com/2014/07/01/opinion/jaron-lanier-on-lack-of-transparency-in-facebook-study.html` (visited on Nov. 18, 2015).

Latour, Bruno (2005). *Reassembling the social: an introduction to actor-network-theory*. Oxford University Press.

Laurance, Jeremy, Sarah Henderson, et al. (2014). 'Patient Engagement: Four Case Studies That Highlight The Potential For Improved Health Outcomes And Reduced Costs'. In: *Health Affairs* 33.9, pp. 1627–1634.

Lawson, Mark (2005). 'Berners-Lee on the read/write web'. In: URL: `http://news.bbc.co.uk/1/hi/technology/4132752.stm` (visited on Dec. 3, 2014).

Lea, M. and R. Spears (1991). 'Computer-mediated communication, de-individuation and group decision-making'. In: *International Journal of Man-Machine Studies* 34.2, pp. 283–301.

Lessig, Lawrence (2002). *The Future of Ideas. The Fate of the Commons in a Connected World*. Random House.

Lévy, Pierre (1997). *Collective Intelligence*. Plenum Publishing Corporation.

Levy, Steven (2001). *Hackers: heroes of the computer revolution*. Updated afterword. Penguin Books.

Lialina, Olia (2014). 'The Only Thing We Know About Cyberspace Is That Its 640x480. One Terabyte of Kilobyte Age'. 31th Chaos Communication Congress (31C3). Hamburg. URL: `https://www.youtube.com/watch?v=ucMDGKcOODA` (visited on May 3, 2015).

Linux Foundation (2014). *Amazon Web Services, Cisco, Dell, Facebook, Fujitsu, Google, IBM, Intel, Microsoft, NetApp, Rackspace, VMware and The Linux Foundation Form New Initiative to Support Critical Open Source Projects*. URL: `http://www.linuxfoundation.org/news-media/announcements/2014/04/amazon-web-services-cisco-dell-facebook-fujitsu-google-ibm-intel` (visited on July 17, 2015).

Lovink, Geert (2011). *Networks without a cause: a critique of social media*. Cambridge: Polity.

Macmanus, Richard (2003). *The Read/Write Web*. ReadWrite. URL: `http://readwrite.com/2003/04/19/the_readwrite_w` (visited on Dec. 3, 2014).

Madianou, Mirca and Daniel Miller (2012). *Migration and new media: transnational families and polymedia*. Routledge.

— (2013). 'Polymedia: Towards a new theory of digital media in interpersonal communication'. In: *International Journal of Cultural Studies* 16.2, pp. 169–187.

Madrigal, Alexis C. (2012). 'Bruce Sterling on Why It Stopped Making Sense to Talk About 'The Internet' in 2012'. In: *The Atlantic*. URL: `http://www.theatlantic.com/technology/archive/2012/12/bruce-sterling-on-why-it-stopped-making-sense-to-talk-about-the-internet-in-2012/266674/` (visited on Dec. 13, 2015).

Marcus, George E. (1995). 'Ethnography in/of the World System: The Emergence of Multi-Sited Ethnography'. In: *Annual Review of Anthropology* 24.1, pp. 95–117.

Marcuse, Herbert (1955). *Eros and Civilization: A Philosophical Inquiry into Freud*. Sphere books.

— (1964). *One dimensional man: studies in the ideology of advanced industrial society.* Routledge & Kegan Paul.

Marks, Kevin (2012). *The Antifragility of the Web*. Epeus' epigone. URL: `http://epeus.blogspot.co.uk/2012/12/the-antifragility-of-web.html` (visited on June 14, 2015).

Marquess, Steve (2014). *Of Money, Responsibility, and Pride*. Speeds and Feeds. URL: `http://veridicalsystems.com/blog/of-money-responsibility-and-pride/` (visited on July 17, 2015).

Marradi, Alberto (2005). *Raccontar Storie. Un nuovo metodo per indagare sui valori*. Studi Superiori 475. Carocci Editore.

Marvin, Lee-ellen (1995). 'Spoof, Spam, Lurk, and Lag: the Aesthetics of Text-based Virtual Realities'. In: *Journal of Computer-Mediated Communication* 1.2.

Maxwell, Richard and Toby Miller (2012). *Greening the Media*. Oxford University Press.

Maymounkov, Petar and David Mazières (2002). 'Kademlia: A Peer-to-peer Information System Based on the XOR Metric'. In: 1st International Workshop on Peer-to-peer Systems. URL: `http://pdos.csail.mit.edu/~petar/papers/maymounkov-%20kademlia-lncs.pdf` (visited on Aug. 17, 2015).

McCracken, Grant David (1988). *The long interview*. Sage Publications.

McDonald, Kevin (2015). 'From Indymedia to Anonymous: rethinking action and identity in digital cultures'. In: *Information, Communication & Society* 18.8, pp. 968–982.

Meta Stack Exchange Contributors (2014). *What does it mean for a question to be protected?* URL: `http://meta.stackexchange.com/questions/52764/what-is-a-protected-question` (visited on Oct. 10, 2015).

Milan, Stefania (2013). *Social movements and their technologies: wiring social change*. Palgrave Macmillan.

Miller, Daniel (2008). *The Comfort of things*. Polity.

— (2009). *Stuff*. John Wiley & Sons.

— (2011). *Tales from Facebook*. Polity.

— (2012). 'Social Networking Sites'. In: *Digital Anthropology*. Ed. by Heather A. Horst and Daniel Miller. London: Bloomsbury, pp. 146–164.

Miller, Daniel and Jolynna Sinanan (2014). *Webcam*. John Wiley & Sons.

Miller, Daniel and Don Slater (2000). *The Internet: An Ethnographic Approach*. Berg.

Minar, Nelson and Marc Hedlund (2001). 'A Network of Peers: Peer-to-Peer Models Through the History of the Internet'. In: *Peer-to-Peer: Harnessing the Power of Disruptive Technologies*. Ed. by Andy Oram. O'Reilly Media.

Mollick, Ethan (2014). 'The dynamics of crowdfunding: An exploratory study'. In: *Journal of Business Venturing* 29.1, pp. 1–16.

Morozov, Evgeny (2013a). 'The Meme Hustler'. In: *The Baffler* 22.

— (2013b). *To save everything, click here: the folly of technological solutionism*. PublicAffairs.

Mosco, Vincent (2014). *To the cloud: big data in a turbulent world*. Boulder: Paradigm Publishers.

Moseley, Ben and Peter Marks (2006). 'Out of the tar pit'. In: Software Practice Advancement (SPA). St Neots, UK.

Murthy, Mahesh (2015). *Facebook is misleading Indians with its full-page ads about Free Basics.* LinkedIn Pulse. URL: `https://www.linkedin.com/pulse/facebook-misleading-indians-its-full-page-ads-free-basics-murthy` (visited on Jan. 4, 2016).

Negroponte, Nicholas (1995). *Being digital*. New York: Vintage Books.

Novak, Brennan, Barnaby Walters, et al. (2015). *IndieWebify.Me - a guide to getting you on the IndieWeb*. URL: `http://indiewebify.me/` (visited on May 9, 2015).

Office for National Statistics (UK) (2014a). *JOBS03: Employee jobs by industry*. Office for National Statistics (UK).

— (2014b). *JOBS04: Self-employment jobs by industry*. Office for National Statistics (UK).

Olson, Parmy (2012). *We are Anonymous: inside the hacker world of Lulzsec, Anonymous, and the global cyber insurgency*. New York: Little, Brown and Co.

Open Wireless Movement (2011). *Why We Need An Open Wireless Movement*. Open Wireless Movement. URL: `https://openwireless.org/blog/2011/04/why-we-need-open-wireless-movement` (visited on Dec. 1, 2015).

O'Reilly, Tim (2005a). *Web 2.0: Compact Definition?* O'Reilly Radar. URL: `http://radar.oreilly.com/2005/10/web-20-compact-definition.html` (visited on Mar. 5, 2015).

— (2005b). *What Is Web 2.0 - Design Patterns and Business Models for the Next Generation of Software*. O'Reilly. URL: `http://www.oreilly.com/pub/a/web2/archive/what-is-web-20.html` (visited on Sept. 4, 2008).

— (2006). *Web 2.0 Compact Definition: Trying Again*. O'Reilly Radar. URL: `http://radar.oreilly.com/2006/12/web-20-compact-definition-tryi.html` (visited on Mar. 5, 2015).

Papert, Seymour (1980). *Mindstorms: children, computers, and powerful ideas*. Basic Books.

Pariser, Eli (2011). *The filter bubble : what the Internet is hiding from you*. London: Viking.

Paul, Ryan (2008). 'Firefox to get massive JavaScript performance boost'. In: *Ars Technica*. URL: `http://arstechnica.com/information-technology/2008/08/firefox-to-get-massive-javascript-performance-boost/` (visited on Aug. 27, 2015).

Perry, David (2012a). *Bitcoin Attacks in Plain English*. Coding In My Sleep. URL: `http://codinginmysleep.com/bitcoin-attacks-in-plain-english/` (visited on Jan. 3, 2016).

Perry, John (2012b). *The Art of Procrastination: A Guide to Effective Dawdling, Lollygagging and Postponing*. New York: Workman Publishing Company.

Petersen, Søren Mørk (2008). 'Loser Generated Content: From Participation to Exploitation'. In: *First Monday* 13.3.

Pichai, Sundar and Linus Upson (2008). *A fresh take on the browser*. Official Google Blog. URL: `http://googleblog.blogspot.com/2008/09/fresh-take-on-browser.html` (visited on Aug. 27, 2015).

Pinch, Trevor J. and Wiebe E. Bijker (1984). 'The Social Construction of Facts and Artefacts: Or How the Sociology of Science and the Sociology of Technology Might Benefit Each Other'. In: *Social Studies of Science* 14.3, pp. 399–441.

Pittman, John H. (2015). *The Tyranny of the Minimum Viable Product*. Medium - @john-pittman. URL: `https://medium.com/@jonhpittman/the-tyranny-of-the-minimum-viable-product-fb25e2e57e6e` (visited on Apr. 4, 2015).

Poppendieck, Mary and Tom Poppendieck (2003). *Lean Software Development: An Agile Toolkit*. Boston: Addison Wesley.

Postel, Jon (1981a). *RFC 791 — Internet Protocol*.

— (1981b). *RFC 793 — Transmission Control Protocol*.

— (1982). *RFC 821 — Simple Mail Transfer Protocol*.

Powell, Alison (2008). 'WIFI PUBLICS: Producing community and technology'. In: *Information, Communication & Society* 11.8, pp. 1068–1088.

Powell, Alison and Leslie Regan Shade (2012). 'Community and Municipal Wi-Fi Initiatives in Canada'. In: *Connecting Canadians: Investigations in Community Informatics*. Ed. by Andrew Clement, Michael Gurstein, et al. Athabasca University Press, pp. 183–201.

Preston-Werner, Tom (2010). *Readme Driven Development*. URL: `http://tom.preston-werner.com/2010/08/23/readme-driven-development.html` (visited on Oct. 13, 2015).

— (2013). *Semantic Versioning 2.0.0*. URL: `http://semver.org/` (visited on Apr. 21, 2015).

Ratto, Matt (2005). 'Embedded Technical Expression: Code and the Leveraging of Functionality'. In: *The Information Society* 21.3, pp. 205–213.

Resig, John (2012). *Redefining the Introduction to Computer Science*. URL: `http://ejohn.org/blog/introducing-khan-cs/` (visited on Oct. 18, 2015).

Rheingold, Howard (1991). *Virtual reality*. Summit Books.

— (1993). *The virtual community: homesteading on the electronic frontier*. Addison Wesley.

Roberts, Mary Lou (2000). 'Geocities (A) and (B)'. In: *Journal of Interactive Marketing* 14.1, pp. 60–72.

rota, andrea (2015). *dear Kai*. Letter. URL: `http://blog.offscreenmag.com/post/134783505586/dear-kai-i-would-like-to-say-thank-you-for` (visited on Dec. 8, 2015).

rota, andrea and Andrea Pozzi (2013). 'Free software and progressive development. Towards an epistemology of software for a learning society'. Draft manuscript.

Rule, James B. (2007). *Privacy in peril*. Oxford University Press.

Ruoti, Scott, Jeff Andersen, et al. (2015). 'Why Johnny Still, Still Can't Encrypt: Evaluating the Usability of a Modern PGP Client'. In: *arXiv preprint arXiv:1510.08555*. URL: `http://arxiv.org/abs/1510.08555`.

Salzberg, Maxwell (2010). *Decentralize the web with Diaspora*. URL: `https://www.kickstarter.com/projects/mbs348/diaspora-the-personally-controlled-do-it-all-distr/description` (visited on Nov. 26, 2015).

Sandoval, Marisol (2012). 'A Critical Empirical Case Study of Consumer Surveillance on Web 2.0'. In: *Internet and Surveillance: The Challenges of Web 2.0 and Social Media*. Ed. by Christian Fuchs, Kees Boersma, et al. Routledge, pp. 147–169.

Schutz, Alfred and Thomas Luckmann (1973). *The Structures of the Life-world*. Northwestern University Press.

Shepherd, Richard (2013). 'Jeremy Keith - Interview for Offscreen magazine'. In: *Offscreen Magazine* 4. In collab. with Jeremy Keith, pp. 58–69.

Shora, Andy (2014). *The Myth of the Full-stack Developer*. andyshora.com. URL: `http://andyshora.com/full-stack-developers.html` (visited on Oct. 17, 2015).

Silverstone, Roger (1994). *Television and everyday life*. Routledge.

Silverstone, Roger and Eric Hirsch (1992). *Consuming technologies: media and information in domestic spaces*. Routledge.

Sinha, V.S., S. Mani, et al. (2013). 'Exploring activeness of users in QA forums'. In: 10th IEEE Working Conference on Mining Software Repositories (MSR) (2013), pp. 77–80.

Skinner, Jonathan (2012). *The interview: an ethnographic approach*. Berg.

Skype (2015). *What are P2P communications?* URL: https://support.skype.com/en/faq/FA10983/what-are-p2p-communications (visited on Jan. 3, 2016).

Slater, Don (2013). *New media, development and globalization: making connections in the Global South*. Polity.

Slater, Don and Tomas Aritzía (2009). 'Assembling Asturias: Scaling Devices and Cultural Leverage'. In: *Urban Assemblages: How Actor-Network Theory Changes Urban Studies*. Ed. by Ignacio Farías and Thomas Bender. Routledge, pp. 91–108.

Soghoian, Christopher (2011). *How Dropbox sacrifices user privacy for cost savings*. URL: http://paranoia.dubfire.net/2011/04/how-dropbox-sacrifices-user-privacy-for.html (visited on Nov. 24, 2015).

SourceForge (2015). *SourceForge - About*. URL: http://sourceforge.net/about (visited on Oct. 10, 2015).

Stallman, Richard (2015). *Who does that server really serve?* URL: https://www.gnu.org/philosophy/who-does-that-server-really-serve.html (visited on Dec. 31, 2015).

Stanley, Phiona (2015). 'Writing the PhD Journey(s): An Autoethnography of Zine-Writing, Angst, Embodiment, and Backpacker Travels'. In: *Journal of Contemporary Ethnography* 44.2, pp. 143–168.

StatCounter (2015). *Platform split (Q1 2009 to Q3 2015)*. URL: http://gs.statcounter.com/#all-comparison-ww-quarterly-200901-201503 (visited on July 28, 2015).

Sterling, Bruce (2013). 'What a feeling!' Webstock 2013. URL: http://www.webstock.org.nz/talks/what-a-feeling/ (visited on Jan. 13, 2015).

Stevens, Ted (2006). *A series of tubes*. US Senate hearing, 28 June 2006.

Stoddard, Greg (2015). 'Popularity and Quality in Social News Aggregators: A Study of Reddit and Hacker News'. In: *arXiv:1501.07860 [physics]*. arXiv: 1501.07860.

Suchman, Lucy (2013). 'Consuming Anthropology'. In: *Interdisciplinarity: Reconfigurations of the Social and Natural Sciences*. Ed. by Andrew Barry and Georgina Born. Routledge.

Sverdlik, Yevgeniy (2014). *French Web Host Builds Bare Metal ARM Server Cloud*. Data Center Knowledge. URL: http://www.datacenterknowledge.com/archives/2014/

11/19/french-web-host-builds-bare-metal-arm-server-cloud/ (visited on Jan. 4, 2016).

Swan, Melanie (2015). *Blockchain*. O'Reilly Media.

Taylor, Charles (2004). *Modern Social Imaginaries*. Durham: Duke University Press.

Terranova, Tiziana (2004). *Network culture: politics for the information age*. Pluto Press.

Tifferet, Sigal and Iris Vilnai-Yavetz (2014). 'Gender differences in Facebook self-presentation: An international randomized study'. In: *Computers in Human Behavior* 35, pp. 388–399.

Torvalds, Linus (2015). *CodeOfConflict*. URL: https://git.kernel.org/cgit/linux/kernel/git/torvalds/linux.git/commit/?id=b0bc657 (visited on July 28, 2015).

Turing, Alan Mathison (1936). 'On computable numbers, with an application to the Entscheidungsproblem'. In: *Proceedings of the London Mathematical Society* s2-42.1, pp. 230–265.

Turkle, Sherry (1995). *Life on the Screen: Identity in the Age of the Internet*. New York: Simon and Schuster.

— (2011). *Alone together: why we expect more from technology and less from each other*. New York: Basic Books.

Turner, Fred (2006). *From counterculture to cyberculture: Stewart Brand, the Whole Earth Network, and the rise of digital utopianism*. Chicago: University of Chicago Press.

Turow, Joseph, Michael Hennessy, et al. (2015). *The tradeoff fallacy - How Marketers Are Misrepresenting American Consumers And Opening Them Up to Exploitation*. Annenberg School for Communication, University of Pennsylvania.

UK Government Migration Advisory Committee (2015). *Partial review of the Shortage Occupation Lists for the UK and for Scotland*. URL: https://www.gov.uk/government/uploads/system/uploads/attachment_data/file/406775/Partial_review_of_the_SOL_for_UK_and_Scotland_Report.pdf (visited on July 28, 2015).

US Congress (1992). *Scientific and Advanced-Technology Act of 1992.* 102nd session, section 1146, sponsored by Barbara Mikulski.

Vaidhyanathan, Siva (2004). *The anarchist in the library : how the clash between freedom and control is hacking the real world and crashing the system*. New York: Basic Books.

Van Dijck, J. and D. Nieborg (2009). 'Wikinomics and its discontents: a critical analysis of Web 2.0 business manifestos'. In: *New Media & Society* 11.5, pp. 855–874.

Vance, Ashlee (2011). 'This Tech Bubble Is Different'. In: *Bloomberg Business*. URL: http://www.bloomberg.com/bw/magazine/content/11_17/b4225060960537.htm (visited on Apr. 22, 2015).

Vega, Davide, Llorenc Cerda-Alabern, et al. (2012). 'Topology patterns of a community network: guifi.net'. In: *Wireless and Mobile Computing, Networking and Communications (WiMob), 2012 IEEE 8th International Conference on*. IEEE, pp. 612–619.

Venners, Bill (2004). *The Simplest Thing that Could Possibly Work*. In collab. with Ward Cunningham. URL: http://www.artima.com/intv/simplest.html (visited on Apr. 18, 2015).

Vergelis, Maria, Tatyana Shcherbakova, et al. (2015). *Kaspersky Security Bulletin. Spam in 2014*. Kaspersky Labs. URL: http://securelist.com/analysis/kaspersky-security-bulletin/69225/kaspersky-security-bulletin-spam-in-2014/ (visited on Apr. 22, 2015).

Victor, Bret (2012a). 'Inventing on principle'. Conference keynote. Canadian University Software Engineering Conference. Montréal. URL: http://vimeo.com/36579366 (visited on Apr. 12, 2015).

— (2012b). *Learnable Programming*. URL: http://worrydream.com/LearnableProgramming/ (visited on Oct. 13, 2015).

Villa, Luis (2013). *Pushing back against licensing and the permission culture*. URL: http://lu.is/blog/2013/01/27/taking-post-open-source-seriously-as-a-statement-about-copyright-law/ (visited on Jan. 16, 2015).

W3C (1999). *HTML 4.01 Specification*.

— (2007). *HTML Design Principles, W3C Working Draft 26 November 2007*. URL: http://www.w3.org/TR/html-design-principles/ (visited on July 26, 2015).

Wachs, Matthias (2015). 'A Secure and Resilient Communication Infrastructure for Decentralized Networking Applications'. Doctoral dissertation. München: Technische Universität München. URL: https://gnunet.org/sites/default/files/NET-2015-02-1.pdf (visited on Sept. 28, 2015).

Walsh, Jo (2006). *Re: nodel-consumotronic code*. E-mail. Personal communication.

Weaver-Hightower, Marcus B. (2012). 'Waltzing Matilda An Autoethnography of a Father's Stillbirth'. In: *Journal of Contemporary Ethnography* 41.4, pp. 462–491.

Weiss, Brandon (2012). *GitHub Is Your Resume Now*. Anti-pattern. URL: `http://anti-pattern.com/github-is-your-resume-now` (visited on July 28, 2015).

Wenger, Etienne (1999). *Communities of Practice: Learning, Meaning, and Identity*. Cambridge University Press.

WHATWG (2015). *FAQ*. WHATWG Wiki. URL: `https://wiki.whatwg.org/wiki/FAQ` (visited on July 26, 2015).

Whitten, Alma and J.D. Tygar (1999). 'Why Johnny Can't Encrypt: A Usability Evaluation of PGP 5.0'. In: *Proceedings of the 8th USENIX Security Symposium*. Washington, DC, pp. 169–184.

Woolgar, Steve, ed. (2002). *Virtual society?: technology, cyberbole, reality*. Oxford University Press.

Xue, Yi and Richard C. Larson (2015). *STEM crisis or STEM surplus? Yes and yes*. U.S. Bureau of Labor Statistics.

Yahoo! (2015). *Pipes End-of-life Announcement*. Yahoo! Pipes Blog. URL: `http://pipes.yqlblog.net/post/120705592639/pipes-end-of-life-announcement` (visited on June 21, 2015).

Zuboff, Shoshana (2015). 'Big Other: Surveillance Capitalism and the Prospects of an Information Civilization'. In: *Journal of Information Technology* 2015.30, pp. 75–89.

Zuckerberg, Mark (2011). 'F8 2011 Keynote'. Facebook F8. URL: `http://original.livestream.com/f8live/video?clipId=pla_0b68074c-8f61-47bd-9348-f41bafc59c25` (visited on June 21, 2015).