

INDEPENDENT TOGETHER: BUILDING AND MAINTAINING VALUES IN A
DISTRIBUTED WEB INFRASTRUCTURE

by

Jack Jamieson

A thesis submitted in conformity with the requirements
for the degree of Doctor of Philosophy
Graduate Department of Information
University of Toronto

© Copyright 2021 by Jack Jamieson

Abstract

Independent Together: Building and Maintaining Values in a Distributed Web Infrastructure

Jack Jamieson

Doctor of Philosophy

Graduate Department of Information

University of Toronto

2021

This dissertation studies a community of web developers building the IndieWeb, a modular and decentralized social web infrastructure through which people can produce and share content and participate in online communities without being dependent on corporate platforms. The purpose of this dissertation is to investigate how developers' values shape and are shaped by this infrastructure, including how concentrations of power and influence affect individuals' capacity to participate in design-decisions related to values. Individuals' design activities are situated in a sociotechnical system to address influence among individual software artifacts, peers in the community, mechanisms for interoperability, and broader internet infrastructures.

Multiple methods are combined to address design activities across individual, community, and infrastructural scales. I observed discussions and development activities in IndieWeb's online chat and at in-person events, studied source-code and developer decision-making on GitHub, and conducted 15 in-depth interviews with IndieWeb contributors between April 2018 and June 2019. I engaged in critical making to reflect on and document the process of building software for this infrastructure. And I employed computational analyses including social network analysis and topic modelling to study the structure of developers' online activities.

This dissertation identifies how values of import to IndieWeb's community are employed in designing its material architectures as well as community policies. This includes an ongoing balance between supporting individuals' agency over personal design decisions and a need to maintain commensurability for the sake of interoperability. In many cases, early decisions about this balance have contributed to barriers for certain types of participants. Yet, those who can cross those barriers experience a lack of stabilization in IndieWeb's infrastructure as a means of achieving richer engagements with technology. By studying design activities as longitudinal and situated within broader infrastructures, this dissertation describes how changing situations and a variety of influences affect possibilities for articulating values through material engagement, offering insights about how to support positive and healthy relationships with technology.

Acknowledgements

I'm extremely grateful to have been surrounded by supportive colleagues, friends, and family during this dissertation.

First, thank you to IndieWeb's community. David Shanske made Yarns possible and taught me to be a better developer. Tantek Çelik was extremely generous with his time sharing nuances of IndieWeb's history. Writing with Greg McVerry highlighted the role of learning in IndieWeb's community. Ryan Barrett provided meticulous feedback about Bridgy's technical details. And too many others to name everyone. Thank you to all who participated in an interview, shared their insights on IRC and at events, or otherwise helped make this research such a joy to work on.

My committee members, Nicole Cohen and Matt Ratto, have been extremely generous with their insights. We have shared many conversations that led me to new questions and approaches, were always enjoyable, and ultimately made me a stronger researcher. And of course, thank you to my supervisor, Rhonda McEwen. In every respect, your mentorship has been a perfect fit. Every time we talked through some methodological detail or other aspect of this research, I've emerged with more confidence, curiosity, and motivation.

Thank you to the many other mentors and colleagues who have supported me along the way. Jeffrey Boase has been constant and generous collaborator and I owe him an inexpressible debt of gratitude. Naomi Yamashita's dedication to using design research to help the disadvantaged is a major inspiration on my work. Jan Hadlaw, who supervised my MA, introduced me to STS and related approaches for thinking about culture and technology. Dawn Walker's insights about decentralization have been a strong influence on my thinking. Hervé Saint-Louis has been a close confidant during every stage of my PhD. Emily Maemura provided valuable feedback on early drafts. Peter Unwin has been a dear friend and colleague from day one of grad school. The "McEwen Squad"—Michelle Lui, Zhao Zhao, and Yaxi Zhou—formed a strong support network for the final stretch. And I am privileged to have learned from Curtis, Dan, Gabby, Kaveh, Rekha and everyone else with whom I've enjoyed long chats at Semaphore, as well as my cohort and the rest of the iSchool community.

Finally, thank you to Elena, whose love and support mean the world; the Lundys, who are a second family; Elizabeth, who supports me even though I used to make way too much noise while she was watching Buffy; and my parents, for everything.

This dissertation was supported by a Social Science and Humanities Research Council Canada Graduate Scholarship, 2015-2018.

Contents

1	Introduction	1
1.1	Motivation and purpose	1
1.2	Background	2
1.2.1	Building for Web 1.0	3
1.2.2	Web 2.0	4
1.2.3	Decentralized Web	6
1.2.4	About the IndieWeb	9
1.2.5	Personal standpoint	11
1.3	Contribution	12
1.4	Structure of the Dissertation	13
2	Background and Literature Review	14
2.1	Questions of values and technology	14
2.1.1	Defining and Locating Values	16
2.2	Infrastructures and platforms	19
2.3	Political Economy	22
2.4	Labour and technological communities of practice	25
2.4.1	Technological divisions of labour	26
2.4.2	Learning in communities of practice	27
2.5	Theoretical approach and Research Questions	29
3	Research Design	32
3.1	Research setting	33
3.2	Participant-observation	35
3.2.1	Critical making	36
3.2.2	Interviews	39
3.2.3	Coding and Reflecting on Field Notes	41
3.3	Logged interactional data	42
3.3.1	Collecting and Cleaning Log Data.	43
3.3.2	Analysis: Topic Modelling	46
3.3.3	Analysis: Network Analysis	49
3.3.4	Linking chat and GitHub data	51
3.4	Combining qualitative and computational analyses	54
3.5	Ethical considerations	55

3.6	Limitations	57
4	Building for yourself: Creating IndieWeb tools with values in mind	59
4.1	Introduction	59
4.2	Autonomy and self-empowerment as guiding values	60
4.2.1	Empowerment and making	61
4.2.2	Autonomy and Free Software	62
4.3	Principles of building on the IndieWeb	63
4.3.1	Why and what to build on the IndieWeb	65
4.3.2	How to build on the IndieWeb	67
4.4	Coordination mechanisms	70
4.4.1	Technical standards / Building blocks	72
4.5	Building Yarns	77
4.5.1	Motivations and background	78
4.5.2	From <i>Yarns Indie Reader</i> to <i>Yarns Microsub Server</i>	79
4.5.3	Plurality in IndieWeb sites	83
4.5.4	Distributed collaboration and Microsub data structures	85
4.6	Discussion and conclusion	89
4.6.1	Reflection and balancing individual with communal values	89
4.6.2	Employed values	92
4.6.3	Conclusion	94
5	Mapping IndieWeb’s community structure with logged data analysis	95
5.1	Summary statistics about IndieWeb’s Chat	96
5.1.1	Length of chat participation	97
5.2	Results: Topic Modelling	98
5.2.1	Topic probability distribution across documents	98
5.2.2	Distribution of keywords across topics	99
5.2.3	Labelling and grouping topics	101
5.2.4	Results: cluster analysis of topic trends over time	107
5.2.5	Topics over time	109
5.3	Summary statistics about IndieWeb’s GitHub network	111
5.3.1	Event types by year	113
5.3.2	Distribution of commits and issues among repos	113
5.4	Does chat activity relate to GitHub activity?	116
5.4.1	Linking between GitHub and chat	116
5.4.2	Cluster analysis of GitHub activities	117
5.4.3	Connecting chat activity to commit-focused and issue-focused GitHub activity	120
5.5	Limitations	124
5.6	Discussion and Conclusion	124

6	Accounts of community from the ground: Investigating influence and exclusivity	126
6.1	Overview of interview participants	127
6.2	Influence and exclusion around technical experience	129
6.2.1	Influence through technical implementation	129
6.2.2	Formal technical barriers	131
6.2.3	Informal technical barriers	132
6.3	Influence and exclusion around cultural diversity	135
6.3.1	Gender gap	135
6.3.2	Addressing exclusion at a community level	139
6.4	Discussion and conclusion	141
7	Building Alongside the Corporate Web	144
7.1	Introduction	144
7.2	Using “tools of our enemies”	145
7.3	Syndicating to Platforms with Bridgy	146
7.4	Data analysis	147
7.5	Results	149
7.5.1	Mapping between URLs and API IDs	149
7.5.2	Mapping privacy from front-end to back-end	151
7.5.3	Precarity and API updates	152
7.6	Discussion and conclusion	154
8	Discussion	157
8.1	RQ1 and RQ2	157
8.2	RQ3	160
8.2.1	Technical ties	160
8.2.2	Human ties	161
8.2.3	Postsocial ties	162
8.3	Working across scales and methods	163
8.4	Independent Together	164
8.4.1	Investment as a virtue	167
8.4.2	Community as part of design	168
8.4.3	Conclusion	170
9	Conclusion	172
9.1	Contributions	172
9.1.1	Values in HCI and design	172
9.1.2	Decentralized Web and alternative social media	173
9.1.3	Logged social media data analysis	173
9.1.4	IndieWeb	174
9.2	Future directions	174
	Appendices	176
A	Interview Invitation	177

B	Interview Consent Form	179
C	Interview Protocol	181
D	Procedure for identifying IndieWeb-related repositories on GitHub	186
E	Keywords used to validate LDA topic models	188
	E.1 Principles (strict list)	188
	E.2 Principles (fuzzy list)	188
	E.3 IndieWeb related standards	190
	E.4 IndieWeb events	190
	E.5 IndieWeb-related applications/software	191
F	Procedure for selecting ‘Principle-related’ keywords	194
G	Timeline of IndieWeb events by location	195
H	Top terms for 16 topic model	196
I	Proportion of chat clusters related to each topic over time (observed).	198
J	Network graphs of GitHub network by year	199
	J.1 2011	199
	J.2 2012	200
	J.3 2013	201
	J.4 2014	202
	J.5 2015	203
	J.6 2016	204
	J.7 2017	205
	J.8 2018	206
	J.9 2019	207
	J.10 All-time	208
K	Mann-Whitney results	209
	K.1 GitHub repo degree centrality by location of IndieWeb keywords	209

List of Tables

3.1	IndieWeb events attended during my research.	37
3.2	Basic demographics of interview invitations and participants	40
3.3	Types of events in IRC archives	44
4.1	Partial list of IndieWeb technical standards with definitions.	73
4.2	Multiple structures for structuring feeds in MF2.	84
5.1	Overview of chat log data	96
5.2	Count of messages per user posted to IndieWeb Chat	97
5.3	Number of active months per chat user	97
5.4	Logistic Regression predicting to identified GitHub activity among IndieWeb chat participants	117
5.5	Logistic Regression predicting to <i>commit-focused</i> GitHub activity	122
5.6	Logistic Regression predicting to <i>issue-focused</i> GitHub activity	123
6.1	Demographic data about interview participants.	128
6.2	Comparison of chat activity among interview participants and other chat users.	129
7.1	Types of formats for Facebook IDs	149
H.1	Top terms for 16 topic model. erated from pyLDAVis with $\lambda = 0.6$	196
K.1	Mann-Whitney U-test results: GitHub repo degree centrality by location of IndieWeb keywords.	209

List of Figures

1.1	Centralized, decentralized, and distributed network structures. Figure from (Baran 1964, p. 2).	8
3.1	Comparison of coherence scores for different pooling methods.	47
3.2	Topic model coherence by topic count.	48
4.1	IndieWeb principles from <i>indieweb.org/principles</i> . Screen captured on January 29, 2020.	64
4.2	Screenshot of <i>Yarns Indie Reader</i> .	80
4.3	Left: Screenshot of Yarns Microsub Server interface for organizing subscriptions into channels. Right: Viewing feeds in a Microsub client called Together (https://alltogethernow.io).	81
5.1	Number of messages per month	96
5.2	Number of unique users per month	96
5.3	Distribution of dominant topic weights per document.	98
5.4	Distribution of principles-related keywords across topics. <i>Left</i> : Strict list of principles-keywords. <i>Right</i> : Fuzzy list of principles keywords.	99
5.5	Distribution of standards-related keywords across topics.	100
5.6	Distribution of IndieWeb project related keywords across topics.	100
5.7	Distribution of event related keywords across topics.	100
5.8	Hierarchical dendrogram of topic clustering results. The red line indicates the cutoff point used to determine the number of clusters (7).	108
5.9	Radar plots showing weight of normalized topic shares per cluster. Every tenth percentile is overlaid on top of one another in transparent blue. Light (transparent) shades indicate values at high percentiles, and darker (more opaque) shades indicate values at lower percentiles.	109
5.10	Violin charts showing distribution of topic shares in each cluster. X-axis shows topic share normalized to a range of -1.00 to 1.00.	110
5.11	Proportion of chat clusters related to each topic over time.	111
5.12	IndieWeb-related GitHub repositories with at least one event by year.	112
5.13	People who made at least one contribution to IndieWeb-related repositories by year.	112
5.14	Github actions by year.	113
5.15	Distribution of users per repository by action category. <i>Left</i> : Including Outliers. <i>Right</i> : Outliers removed.	114

5.16	Network chart of IndieWeb-related GitHub repos and users from Jan 1, 2019 through May 17, 2019.	115
5.17	Hierarchical dendrogram of GitHub activity clustering results.	118
5.18	Radar plots showing weight of GitHub actions per cluster. Every tenth percentile is overlaid on top of one another in transparent blue. Light (transparent) shades indicate values at high percentiles, and darker (more opaque) shades indicate values at lower percentiles.	118
5.19	Violin charts showing distribution of topic group values in each cluster. X-axis shows topic share normalized to a range of 0.00 to 1.00.	119
5.20	Scatter plot showing distribution of commit and issue scores among two clusters.	119
G.1	Timeline of IndieWeb events by location. Reproduced from https://indieweb.org/IndieWebCamps as of May 24, 2020.	195
I.1	Proportion of chat clusters related to each topic over time.	198
J.1	Network chart of IndieWeb-related GitHub repos and users during 2011.	199
J.2	Network chart of IndieWeb-related GitHub repos and users during 2012.	200
J.3	Network chart of IndieWeb-related GitHub repos and users during 2013.	201
J.4	Network chart of IndieWeb-related GitHub repos and users during 2014.	202
J.5	Network chart of IndieWeb-related GitHub repos and users during 2015.	203
J.6	Network chart of IndieWeb-related GitHub repos and users during 2016.	204
J.7	Network chart of IndieWeb-related GitHub repos and users during 2017.	205
J.8	Network chart of IndieWeb-related GitHub repos and users during 2018.	206
J.9	Network chart of IndieWeb-related GitHub repos and users from Jan 1, 2019 through May 17, 2019.	207
J.10	Network chart of IndieWeb-related GitHub repos and users from Jan 1, 2011 through May 17, 2019.	208

Chapter 1

Introduction

1.1 Motivation and purpose

In this dissertation, I study a community of Web developers attempting to build tools to support a decentralized social Web. This is a Web in which people can produce and share content and participate in online communities without relying on corporate platforms. Efforts to build alternatives to centralized Web platforms are rooted in a belief that communication technologies are connected to social power. Currently, large Internet companies exhibit significant cultural, social, and political influence, and it is vital to study how alternatives might offer new arrangements of power. I investigate the activities involved in building and maintaining one such alternative, with particular attention to the role of designers' values. I will define "values" further in Section 2.1.1 but briefly, here I am referring to designers' commitments to support "human well-being, human dignity, justice, welfare, and human rights" (Friedman and P. H. Kahn 2003, p. 1186).

Consider what such a system might look like. People would need a place to host their Web presence, and, to avoid any individual company from having too much power, there would be many different options for Web hosting. If this were to be a 'social' Web, then activities such as posting original content and commenting on, sharing, and 'liking' other people's posts would be simple and engaging. To avoid relying on intermediary platforms, communication would pass directly from individual to individual. And to emphasize individuals' control over their own online presence, it could be possible to exchange messages among people even if they use different software or Web hosts from each other. The vision I am describing integrates social features that have become common in social media platforms with an approach to hosting content that resembles the Web of the 1990s and early 2000s. In this period of the Web, personal websites relied largely on commercial services for hosting, but there were a variety of choices, which mitigated the possibility of a small number of companies to dominate globally to the extent that is possible today. Furthermore, personal websites were profoundly customizable and demonstrated significant diversity of content, structure, and style, especially when compared to the still expressive but much more standardized structures of content on large social media platforms.

In describing this vision, I draw upon the creator of the Web, Tim Berners-Lee's, account that the Web was designed to be decentralized, with "no central computer 'controlling' the Web, no single network on which these protocols worked, not even an organization anywhere that 'ran' the Web." (Berners-Lee and Fischetti 1999, p. 36). My main site for analysis is a community of developers building the IndieWeb,

a system that operates generally along the logics I have just described. Importantly, there is no IndieWeb “platform,” but instead it is a network of individual websites connected using technical standards. It is built by a large international community, and almost all IndieWeb builders do so as a hobby rather than as a career. While individual contributors’ motivations vary significantly, commitments to values such as individual autonomy, self-empowerment, and openness are common. As I study IndieWeb’s development, I am particularly concerned with how such values are employed in design activities and articulated in resulting technical structures.

This dissertation’s concern with relationships among human values, work practices, and technical artifacts demonstrates a perspective of IndieWeb as a sociotechnical system. From this perspective, society and technology are co-constitutive; that is, structures of human relationships simultaneously influence and are influenced by the development, maintenance, and use of technology. Thus, as I investigate the construction and maintenance of an emerging Web infrastructure, one of my main motivations is to identify influences among individuals’ design activities, communities in which those activities are coordinated, and larger infrastructures into which emerging Web systems are embedded. To accomplish this, I combine close technical and cultural inquiries, computational analyses, and political economic perspectives of labour and power. Additionally, I employ a longitudinal lens, which allows me to describe how decisions relate to one another across time. By studying design activities as longitudinal and situated within broader infrastructures, this dissertation describes how changing situations and a variety of influences affect possibilities for articulating values through material engagement and collaborative structures. As a result, this dissertation contributes to knowledge about studying platforms, infrastructures, and distributed collaboration.

1.2 Background

This section describes the context of this study. I begin by reviewing historical narratives about building for the Web, focusing on the transition from an apparently independent Web in the 1990s to the domination of platforms in the 2010s. This account is bounded by attention to how the structure of the Web shaped activities involved in *individuals’* online experience; this means I do not discuss larger scale sites such as the businesses that fuelled the dot-com bubble, except to discuss how they shaped possibilities for individuals to use the Web. This scope highlights relevant factors for understanding the IndieWeb, since it is oriented primarily around individuals’ websites.

An additional consideration while reading this account is that I concentrate on dominant narratives and overlook debates and perspectives from margins. I accept a view of the early Web as something like a mythology, adopting a lens like that of Mosco in *The Digital Sublime*, in which he writes, “Myths are not true or false, but living or dead” (p. 3). This does not mean I accept Silicon Valley and other dominant tech maxims as truth, but in this description, I focus on the ways these accounts form a philosophical foundation for projects like IndieWeb.

Following this historical review, I introduce the landscape of efforts to build decentralized alternatives to centralized Web platforms. Finally, I present an overview of the IndieWeb, including key technical standards and structures, its community, and its relationship to the Web more broadly.

1.2.1 Building for Web 1.0

When Berners-Lee proposed the World Wide Web in 1989 it was conceived as a system for document sharing. The earliest websites consisted of static documents formatted using HTML, a markup language Berners-Lee created based on SGMLguid, an SGML (Standard Generalized Markup Language) format that was used at CERN, where Berners-Lee worked. Berners-Lee describes HTML's similarities to SGMLguid as a diplomatic choice. By doing so, he could seek support from a large community who used SGML, and furthermore colleagues at CERN who were familiar with SGMLguid would be able to quickly understand how they could use HTML (Berners-Lee and Fischetti 1999). Each HTML document has a fixed address called a URI (Universal Resource Identifier), and hyperlinks allow one document to point to another's URI, so Web users can navigate between these documents.

Decentralization was an explicit goal in the initial design of the Web; Berners-Lee recalls a driving motivation for the Web's architecture: "The system had to have one other fundamental property: It had to be completely decentralized. That would be the only way a new person somewhere could start to use it without asking for access from anyone else." (Berners-Lee and Fischetti 1999, p. 16). He proceeds to explain the motivation for decentralization here was scalability, "I wanted the act of adding a new link to be trivial; if it was, then a Web of links could spread evenly across the globe" (1999, p. 16).

The Web became more complex as it was adapted for a larger variety of uses. The ability to display images was implemented in the Mosaic Web browser and then became a standard in HTML 2.0 (Raggett et al. 1998). In 1995 JavaScript was created, which allows program code to be run within websites. Cascading Style Sheets allowed for more complex visual layouts of webpages. Server-side programming allowed for webpages generated on the fly using content from a back-end database. And AJAX brought dynamism to the front-end of webpages by allowing them to fluidly load new content in real-time.

In addition to providing a richer user experience, Web design moved toward increased standardization. HTML became an open standard, published through the Internet Engineering Task Force (IETF) beginning in 1993 (Berners-Lee and D. W. Connolly 1995) and then through the World Wide Web Consortium (W3C) from January 1997 onward (Raggett 1997). One of the goals of standardizing HTML and related Web protocols was to ensure Web pages looked and behaved the same on different browsers—a considerable challenge amidst the increasingly complex multimedia capabilities of the Web. Additionally, there was a growing concern with making the Web accessible for people with disabilities, evidenced by the W3C's establishment of the Web Accessibility Initiative in 1997 (Kennedy 2011).

Alongside the development of standards, another approach for building websites with complex media features was the use of WYSIWYG (What You See Is What You Get) editing software, which became popular in the late 1990s. Instead of writing code and then viewing the resulting website in a browser, designers using WYSIWYG software worked with a visual interface of menus, toolbars, and a preview of the page(s) they were constructing. These tools made it easier for non-coders to build and edit websites, but the code they produced was often poorly formatted and had poor accessibility for people with disabilities. Kennedy (2011) commented on the ethical impact of these editors for professional Web design work:

They were also seen to detract from the caring, craft-like approaches to web design that some practitioners were developing and advocating. It was much harder to be dedicated to doing a job well, to pursue excellence and to benefit from some of the other internal rewards of web design practice with tools that prohibited doing good work. (p. 59)

Whereas the adoption of standards aligned with a shift toward professionalization of Web design, WYSIWYG editors and other easy-to-use tools aimed to make Web design more accessible for amateurs. These represent two distinct approaches for addressing the increasing complexity and ubiquity of the Web: Technical standards enhanced designers' depth of control at the expense of ease of use; and easy-to-use Web editors made Web design easier for non-experts, while sacrificing depth of control.

Through this period, the majority of websites were collections of documents. As a result, hosting was relatively simple; for basic websites, one could simply copy a collection of HTML files and images to a server. Early Web users probably remember being able to create websites using services such as GeoCities, perhaps being provided webspace by their Internet Service Provider, or, by 1999, signing up for early blog services like LiveJournal and Blogger. In sum, although the tools used to build Websites became more sophisticated, the Web remained open in the sense that anyone could create a page without having to ask permission from a central authority.

1.2.2 Web 2.0

Participating in the creation and distribution of Web content became dramatically easier with the advent of Web 2.0 in the mid-2000s. This marked the rise of platforms as the dominant structure of the Web. The transformation between the early Web and Web 2.0 has been described as one from “‘web-as-information-source’ to ‘web-as-participation-platform’” (Song 2010, p. 251). The titans of Web 2.0 are companies like Google, Facebook, and Twitter. These platforms have been celebrated for lowering the level of resources and technical knowledge required to ‘participate’ online (Shirky 2008; Schäfer 2011). However, in the process they have grown into near monopolies with business models oriented around commodifying users' information (i.e., transforming users' content and information about their online activity into data that can be exploited as a resource, such as for targeted advertising) (McChesney 2013; Fuchs 2012; Srnicek 2017).

Before proceeding further, it is necessary to establish a clear definition of centralized Web platforms. Gillespie's (2010) discussion of platform politics describes platforms as “content intermediaries” that host user-generated content (p. 348). This intermediary position is significant—platforms tend to be centralized insofar as they occupy a central place in networks of actors, and it is from this position that “platforms gain not only access to more data but also control and governance over the rules of the game” (Srnicek 2017, p. 52). Gillespie distinguishes between the computational dimension of platforms—their capacity to be built upon with software and data—and more figurative dimensions that serve a variety of actors such as creators, audiences, and marketers. This supports Evans, Hagi, and Schmalensee's (2006) argument that the most critical economic feature of software platforms is “their potential for supporting a multisided business—one in which value is created by bringing together on the same platform multiple distinct groups of customers who need each other in some way” (p. 3). van Dijck, Poell, and de Wall's (2018) definition of platforms covers similar territory, and further argues that platform architectures tend to be “geared towards the systematic collection, algorithmic processing, circulation, and monetization of user data” (p. 4). Finally, the IndieWeb community, which is the focus of this dissertation, is especially concerned with platforms that enclose data in some way (which they call “silos”):

A silo, or web content hosting silo, in the context of the IndieWeb, is a centralized web site typically owned by a for-profit corporation that stakes some claim to content contributed to it and restricts access in some way (has walls). (IndieWeb.org 2017)

These definitions highlight key qualities of platforms for the purpose of this project, so my working definition of centralized Web platforms is *Web architectures or systems that*:

1. Host user-generated content and organize interactions among users.
2. Can be built upon using third-party software and data.
3. Facilitate multi-sided markets.
4. Occupy a central position in their networks of users and other actors.
5. Profit from user data.
6. Restrict access to data in some way.

This definition is intended to be robust enough to refer to a wide range of systems, while retaining a focus on the centralized ‘silos’ about which IndieWeb is primarily concerned. As suggested by the emphasis this definition places on business models, the platforms to which I am referring are corporate. In fact, IndieWeb positions itself as an alternative to the “corporate web” (IndieWeb.org 2020c). However, rather than using a shorthand like “corporate platforms” I use the term “centralized” to distinguish from new types of decentralized platforms that are sometimes governed by corporate structures but nonetheless which operate under different business models than centralized Web platforms.¹

Centralized Web platforms are important for this dissertation because they form the structures against which decentralized alternatives such as IndieWeb are framed. Centralized Web platforms’ business models employ a double process of commodification (Cohen 2013). First, the content created and circulated by platform users is accompanied by advertisements; Second, the information surrounding this content—e.g. demographic data, interests and habits identified through social media activities—is aggregated and sold for profit and/or used to precisely target advertisements. One of the design strategies proposed for commodifying users’ data was, “Set inclusive defaults for aggregating user data as a side-effect of their use in the application” to achieve “network effects by default” (O’Reilly 2005, p. 5), meaning that a platform’s capacity to generate revenue increases in proportion to its number of users. Helmond (2015) articulated that platforms’ influence extends beyond their own sites and applications, asserting that the Web is undergoing *platformization* defined by a simultaneous decentralization of platform features across the Web and recentralization of individuals’ data to be stored and analyzed on platform servers. This means one does not need to visit a platform to be subject to its monitoring, since social media sharing buttons and other platform elements collect data when they are embedded on third-party websites (see for e.g. Roosendaal 2011).

By enclosing data in “walled gardens” or “silos,” large platforms act as gatekeepers for access. This is a sharp contrast with the original vision for the Web as a network of linked documents, where (1) a document could be accessed by anyone who knew its URI, and (2) anyone could put a document online without having to first seek permission (Berners-Lee 2010). Platforms generally require users to register an account in order to post content, which requires agreeing to a Terms of Service (TOS) agreement. For the most part, TOS are rarely read and do not form a significant barrier for typical users. However,

¹For example, Protocol Labs, the company behind a decentralized file storage system called IPFS (InterPlanetary File System) and an open source cryptocurrency called Filecoin, received \$52 Million in funding from several high-profile venture capital firms who participated in a pre-sale for Filecoin’s ICO (Initial Coin Offering) (Chernova 2017). It is prudent, in my opinion, to recognize this as “corporate” while still distinguishing from the centralized platform business models that have dominated Web 2.0.

there have been many cases where enforcement of TOS has excluded users who fell outside a platform's expectations of 'normal' or 'acceptable' users. For example, Facebook requires users to be identified by their real name and has suspended the accounts of some Native American users whose names contain phrases that Facebook's filter evaluates as fake (Haimson and Hoffmann 2016). Even when content is not removed, platforms rely on algorithms to present users with content they determine to be relevant or engaging for each individual. The dominance of large platforms means that content creators must negotiate with opaque algorithmic filters to promote their content (Wu, Pedersen, and Salehi 2019), further entrenching their gatekeeper status. Finally, Srnicek (2017) describes how platform structures are extending well beyond the Internet and that this raises serious concerns for political economy at a large scale.

Alongside these criticisms, it is important to recognize the utility of platforms. As noted above, the participatory potential of Web platforms has been celebrated in scholarship asserting that Web 2.0 flattened barriers for creating and distributing Web content and so empowered individuals to actively participate online instead of simply consuming what others created (Shirky 2008; Shirky 2010; Bruns 2008). This view was famously espoused in Time Magazine's selection of "you" for person of the year in 2006: "For seizing the reins of the global media, for founding and framing the new digital democracy, for working for nothing and beating the pros at their own game, TIME's Person of the Year for 2006 is you" (Grossman 2006). As a mythological base for the contemporary Web, this remains a powerful argument. Platforms have contributed to the streamlining of producing and publishing various types of media, and this has supported new forms of activism (e.g. Khamis and Vaughn 2011), self-expression (e.g. Senft and Baym 2015), and even careers (e.g. Wu, Pedersen, and Salehi 2019). Atton (2002) described alternative media as being "crucially about offering the means for democratic communication to people who are normally excluded from media production" (p. 4). In this respect, centralized Web platforms have often been presented as an alternative to corporate media (especially broadcast and print media). However, evidenced by the increasing domination of a small number of centralized Web platforms, and ways in which their business models have structured their design, it is clear that "social media's potential challenge to corporate media was absorbed by informational capitalism" (Gehl 2015b, p. 3).

In sum, the platform-centred model of Web 2.0 has been celebrated for creating new opportunities for individuals to participate in shaping their culture. And simultaneously, it has led to concentrations of ownership and control. Both of these dimensions are influences upon efforts to build alternative Web structures. Emerging alternatives generally embrace decentralization as a way to move away from concentrations of power, thus reacting against large Web 2.0 platforms. But simultaneously, the ease-of-use that has been fostered by platforms remains a positive influence on user experience designs.

1.2.3 Decentralized Web

There are many efforts to build alternatives to centralized Web platforms, and almost all are aligned in their agreement with at least some of the criticisms described above, as well as a general commitment to decentralization as a means to redistribute power from platforms to individuals by allowing them to control how they communicate with others online, how they present themselves (see for e.g. Guy 2017), and to own and control their personal data.² These projects often refer to the early Web with reverence. Platforms are described as a move in the wrong direction and calls to "re-decentralize" the Web (Berners-

²The following discussion about decentralized Web projects has been informed by conversations with Dawn Walker, and draws from research we presented at the 2019 *Our Networks* conference in Toronto (Jamieson and Walker 2019).

Lee 2016; Wirdum 2016) express a yearning to return to the Web’s origins. These efforts comprise a variety of approaches, some in competition with one another while others are complementary. Even as they work to contest the power structures of centralized Web platforms, many decentralized Web projects embrace their features and user interface designs. In fact, many of these projects act as alternatives to social media platforms, such as Mastodon and Micro.blog (both can be considered alternatives to Twitter), Diaspora*, which has been framed as a Facebook alternative (Dwyer 2010), and PeerTube (an alternative to platforms like YouTube). Other projects propose radical changes to underlying models for hosting and distributing files on the Internet, such as Interplanetary File System (IPFS), which uses a peer-to-peer model for hosting data instead of the client-server model that has dominated the Web. There are many other approaches and I won’t endeavour to provide a thorough account here. Instead, I wish to convey that there is a breadth of projects attempting to build decentralized Web architectures, and that they are generally framed as individually empowering alternatives to existing systems.

The founder of the Internet Archive, Brewster Kahle, argued for a decentralized Web with an appeal to values:

We need to bake our values into our code. Freedom of expression needs to be baked into our code. Privacy should be baked into our code. Universal access to all knowledge. But right now, those values are not embedded in the Web. (Kahle 2015)

Similarly, Gehl (2015a; 2015b; 2018) has demonstrated that emerging alternative social media are usually motivated to some degree by their creators’ values. The values associated with decentralization are oriented around personal freedom and autonomy. The general mechanism in these systems oriented around the degree to which a single organization or system can control the rest of the network. Descriptions of decentralized networks often present some variation of Paul Baran’s (1964) illustration of centralized, decentralized, and distributed networks, which is presented in Figure 1.1. The leftmost chart in this figure presents a centralized network in which a single node can be an arbiter of the entire system. By contrast, the decentralized and distributed networks lack a central site of authority. Note, too, that contemporary descriptions of “decentralized” networks encompass both decentralized and distributed architectures as defined by Baran. From this view, there is an intuitive link between decentralized network structures and individual autonomy. However, scholarship about emerging decentralized Web projects has highlighted a need to more substantially investigate how values and politics are articulated through their design.

Schneider (2019) argued that behind the veil of *decentralization* as a guiding concept lurk more substantive debates about how power should operate. Specifically, he asserts that “decentralization in one part of a system consistently coincides with centralization in another part” (2019, p. 16). Halpin (2018) provide a compelling hypothetical example, noting that “a version of the Chinese ‘social credit’ system could easily be built in a decentralized manner using blockchain-based smart contracts” (p. 4). While technically decentralized, such a project would conflict with the values of individual autonomy and personal freedom typically associated with decentralization. Narayanan et al. (2012) investigated failures in decentralized personal data architectures—a specific form of decentralized Web project—and find that values have been conflated with design decisions. They conclude that success in building these systems “is not simply a matter of making some technical decisions” (2012, p. 5) and recommend that designers pay particular attention to how their systems fit within larger regulatory and economic structures, maintain interoperability with related technical systems, and pay attention to the conceptual fidelity of their projects to ensure that their designs address a real need. These examples highlight that

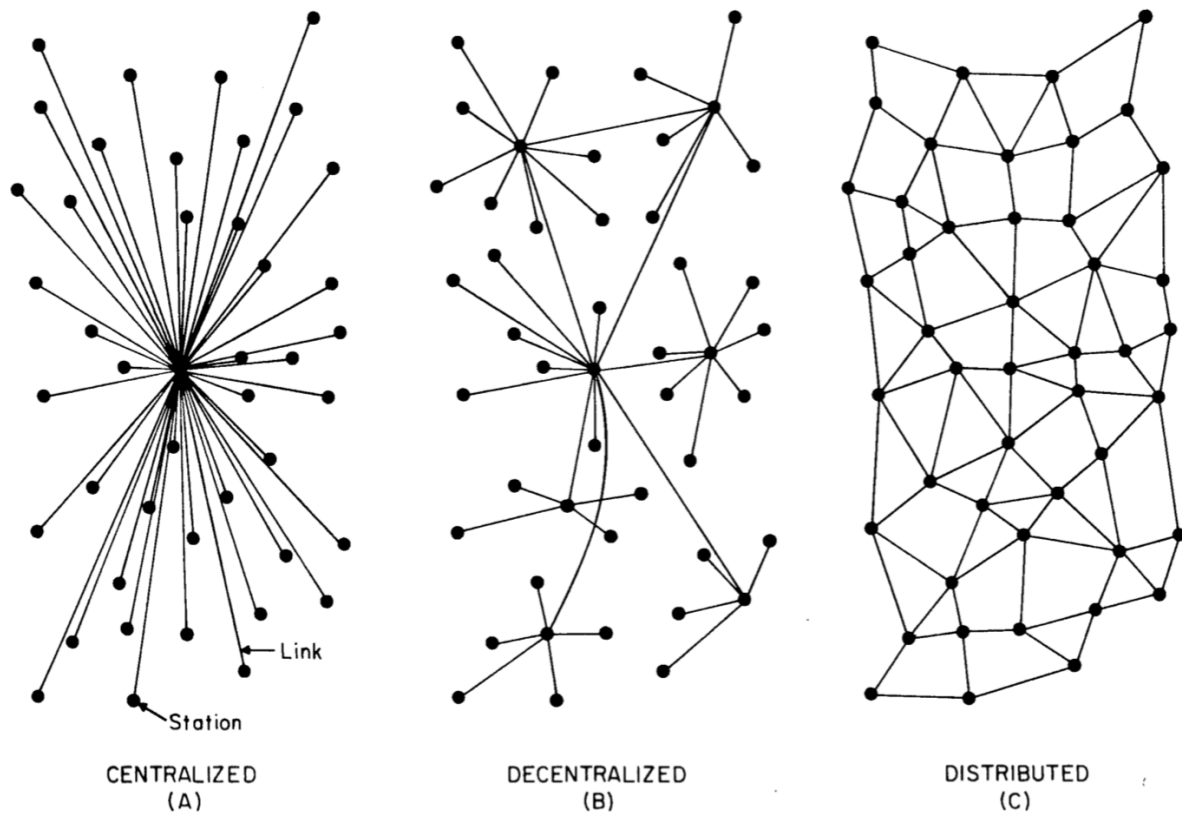


Figure 1.1: Centralized, decentralized, and distributed network structures. Figure from (Baran 1964, p. 2).

any serious consideration of decentralized Web projects as a means to redistribute power must account for relationships between values, design decisions, and broader social and technical forces.

This brief account of the early Web, Web 2.0, and emerging decentralized alternatives has highlighted ways in which building for the Web has changed. Where the early Web was characterized by a strong commitment to decentralization, the rise of platforms has created new centres of power. A variety of projects have emerged that are attempting to reshape the Internet, and there is a gap in scholarship that investigates what is actually involved in building such systems, and how this relates to substantive values of their creators. In the remainder of this section, I present an overview of the IndieWeb, the project this dissertation takes as its main site for analysis.

1.2.4 About the IndieWeb

This dissertation focuses on the design and maintenance of the IndieWeb. In this section, I present a brief description of the IndieWeb and explain why it is a valuable site for my inquiry.

IndieWeb is an international community of people who use personal websites as their primary online identity. This means owning a domain name where they publish blog posts, notes, articles, RSVPs, photos and any other Web content. Relying on personal websites demonstrates an embrace of online practices that characterized the early Web. At the same time, IndieWeb sites embrace conventional social features of Web 2.0, such as *likes*, *sharing* (analogous to retweeting), and social feeds. IndieWeb’s community has defined a number of new standards to allow these websites to engage in these kinds of social activities directly from one personal website to another, with no intermediate platform. Additionally, many IndieWeb sites use platform APIs to distribute their content on platforms, even as its primary home remains on their personal website. This means that one may publish a post to their IndieWeb site, then syndicate it to one or several platforms such as Twitter. Then, when people respond to the syndicated post (e.g. by retweeting or replying) that response can be “backfed” to be displayed on the original post.³ The goal of these methods is for IndieWeb users to benefit from the economies of scale that facilitate platforms’ extensive distribution networks while retaining individual ownership and control over their content.

IndieWeb co-founder Tantek Çelik has framed IndieWeb as a direct response to the dominant position of centralized platforms that emerged during the 2000s (2014). He asserted that large platforms exercise excessive power over users’ online experience and small platforms have proved to be precarious since they frequently go out of business and lose users’ data in the process. Further, few platforms of any size embrace the open standards that would allow individuals to backup or duplicate their data, since platforms rely on a business model centred around user data as a source of profit. As well as responding to platforms, IndieWeb is also influenced by a critique of other technologists’ efforts to decentralize, federate, or otherwise reshape the Web to empower individuals. The concept behind the IndieWeb was developed by Tantek Çelik and Aaron Parecki after they both attended the *Federated Social Web Summit* in 2010. Çelik described that they were frustrated that participants at the summit spent “a lot of time spent on discussing what could be possible” rather than “what *was* possible” (2014). In response, IndieWeb was founded with a focus on building right away. Çelik and Parecki worked with Amber Case and Crystal Beasley to organize the first *IndieWebCamp* in 2011 in Portland, OR, USA. That event consisted of one day of talks and one day of hands-on work sessions. The event was targeted

³This syndication flow is referred to in the IndieWeb community as POSSE (Publish on Own Site, Syndicate Elsewhere). An example of POSSE is discussed further in Chapter 7.

at “builders only” — those who work on code, design, or user-experience for IndieWeb open source projects (IndieWeb.org 2019a).

In subsequent years, IndieWeb steadily increased in size. Since the first IndieWebCamp in 2011, hundreds of IndieWeb events have been held in multiple countries, mostly in North America and western Europe. Further, IndieWeb’s online infrastructure has grown to encompass at least thousands of sites.⁴ Significantly, although there are some services to which one can simply sign up to join the IndieWeb, these co-exist alongside independently operated IndieWeb sites. As a result, IndieWeb does not have a gatekeeping entity, since anyone can “join” the IndieWeb simply by building a website that includes IndieWeb capabilities. (Of course, for those who elect to build their own site, access to Web hosting and know-how to build a site are de-facto requirements). The result is that each IndieWeb site can be distinct, incorporating features that are desirable to individual creators and discarding others. Ultimately IndieWeb’s network of websites is supported by a heterogeneous infrastructure of software, standards, design conventions, social practices, and a large community of contributors who collaborate and chat online and at in-person events.

IndieWeb does not have a formal organizational structure, and the vast majority of participants in this community do so as a hobby. Not everyone who uses IndieWeb is a “builder” in the sense of contributing code, design, or UX to IndieWeb’s infrastructure, but a significant portion are. Since this dissertation is concerned with the *construction* of the IndieWeb, it is not concerned with people who use IndieWeb tools but do not contribute to their development. However, among those who contribute, there is a broad range of participation. Some contributors directly work on technical projects by writing code or contributing to designs. Others may contribute by reporting bugs or offering feedback. And still others may contribute through organizational activities such as event planning and documentation. Contributors across this range have varying backgrounds, from working in the Web industry at prominent organizations such as Mozilla, Medium, or the W3C, to independent developers and hobbyists whose day jobs are unrelated to the IndieWeb.

IndieWeb’s lack of commercialization is of particular import. Almost all IndieWeb sites and services are free and have no revenue stream. The exceptions are two IndieWeb-friendly platforms called *Known* and *Micro.blog*, whose business models relied on subscription fees. Across all IndieWeb sites and services, none that I am aware of feature advertising or otherwise commodify user-data.

Alternative media theory provides some useful scaffolding for characterizing IndieWeb in relation to centralized platforms. As noted earlier, Atton’s (2002) work about alternative media emphasizes their potential to broaden access to democratic communication. Gehl (2015b) summarizes Atton’s point by asserting, “Alternative media are defined as much by their content (i.e., radical, progressive, socialist, anarchist, feminist, queer, or anti-racist) as by the contours and practices of their underlying conditions of production, which are meant to allow democratic participation in making media” (p. 2). IndieWeb focuses on the latter aspect, the conditions of production and, crucially, distribution. Atton’s (2002) *typology of alternative and radical media* is useful for explaining this. This typology includes six features, three of which concern products—(1) content, (2) form, and (3) reprographic innovations/adaptations—and three of which concern processes—(4) distributive use, (5) transformed social relations, roles, and responsibilities, and (6) transformed communication processes (Atton 2002, p. 27). IndieWeb’s overall

⁴It is difficult to develop a precise count of IndieWeb sites, as is the case for many decentralized systems. In 2017, one of IndieWeb’s contributors ran a Web crawl that identified 2300 IndieWeb sites (Barrett 2017), a low estimate given the inherent gaps to crawling methods. A 2016 article on IndieWeb.org asserts that IndieWeb-compatible platform Known had at least tens of thousands of users. And another IndieWeb-compatible platform, *Micro.blog*, was launched in 2018 having been supported by 3,080 backers on Kickstarter (Kickstarter.com 2019).

structure focuses on the latter three features by developing ways for individuals to use their own website rather than rely on centralized Web platforms, to communicate through decentralized linkages among personal websites, and therefore to challenge typical hierarchies between platforms and users. Because each IndieWeb site can be different, it is possible for individuals to demonstrate alternative or radical *products*, but this is not a defining feature of the IndieWeb, and many sites are conventional in that aspect. So, as alternative media, IndieWeb is concerned with transforming processes, and this draws my attention to the work involved in building and maintaining IndieWeb software, websites, technical specifications, and other features. Additionally, Atton (2002) draws on Raymond Williams’ interpretation of alternative and oppositional practices to position alternative media:

Williams made a vital distinction between alternative and oppositional practices. Alternative culture seeks a place to coexist within the existing hegemony, whereas oppositional culture aims to replace it. For instance, there is a world of difference between a minority ‘back-to-nature’ cult and the ecology movement’s global reach. (McGuigan, 1992: 25) (as cited in Atton 2002, p. 19)

IndieWeb occupies a somewhat moderate position in this respect. On one hand, at least at an individual scale, IndieWeb is oppositional in that its purpose is for people to use their personal IndieWeb site instead of relying on centralized Web platforms. On the other hand, several aspects of IndieWeb’s material design and community practices demonstrate coexistence, such as its use of centralized Web platform APIs as a distribution mechanism. Most importantly, IndieWeb does not approach the scale of its corporate counterparts and it is therefore difficult to imagine as an attempt to replace them in totality. In this respect, IndieWeb resembles an anarchic *self-managed* model for alternative media and demonstrates “prefigurative politics” by practicing its political position in the present rather than merely imagining the future (Downing 2001, p. 71).

This overview of the IndieWeb eludes to features that make it a worthwhile site for this dissertation. First, building is a defining characteristic of the IndieWeb, and this is instrumental for understanding IndieWeb as an alternative to centralized Web platforms. From its beginning through the present, IndieWeb contributors are engaged in building the system as they use it. Moreover, IndieWeb has operated steadily since 2011. IndieWeb’s demonstration of building over an extended period makes it a valuable site not only for investigating the initial steps of designing and building, but also the work involved in maintenance. Additionally, IndieWeb is deliberately and openly a heterogenous, diverse, and constantly changing system. IndieWeb’s internal modularity and its repurposing of platform APIs as distribution mechanisms highlight that the Internet is not defined from the top-down, but instead “fixed in modular increments” (Star 1999, p. 382). This presents an in-road for understanding how IndieWeb operates as part of a broader Internet infrastructure, and how individual design decisions may require negotiation to accommodate other aspects of the system.

1.2.5 Personal standpoint

My motivations for this project extend from my experiences working as a Web developer. Prior to graduate school, I worked as a freelance Web developer, enjoying the freedom to be work on my own schedule while still making a good living. Despite a high degree of professional freedom, I was unsure of the moral contribution of my work. Was my work making the world, or even the Web, a better place? I was proud of the technical quality of the sites I built and my clients were happy. Nonetheless, I was

conflicted by frequent requests to implement features such as embedded Twitter feeds, Facebook sharing buttons, and scripts for collecting analytics data. I did not like that these tools were all used to collect and commodify information about Web users, and I'd set up my personal computer to block most of this tracking. It felt hypocritical to make a living by imposing features onto others that I went out of the way to avoid for myself. In short, the explosion of "Web 2.0" helped me earn a good living doing work I enjoyed, but reaping these benefits required me to rely on a variety of third-party platforms whose ethics clashed with my own.

At the same time, I realized the website I was most proud of was one I created for my mother's small business while I was a teenager. Until retiring, my mother designed knitting patterns and taught knitting workshops across North America. I designed and built a website for her that presented her workshop dates, a blog ('knitknotes'), and an online store for ordering her kits and patterns. At a time when I was mostly consumed with high school and a part-time job at a local movie theatre, I was proud to have built a website that could stand alongside the work of professional designers and help support my family. That experience instilled in me a belief that the Web was a powerful vehicle to empower individuals.⁵

These two anecdotes summarize why IndieWeb is compelling for me as a site for inquiry. Its claim to be an alternative to the corporate Web suggests an opportunity to contest some features of the Web with which I disagreed during my freelance career. And, its approach of relying on personal websites that can be operated by individuals evokes the type of empowerment I experienced as a teenager. I highlight this personal standpoint partly as a point of self-reflection, and also to acknowledge that I have a predisposition to be sympathetic with IndieWeb's endeavour, which contributes to my research questions and overall approach.

1.3 Contribution

This dissertation investigates a pivotal moment in the Internet's development. Large corporate platforms have become the dominant structures of the Internet, and their business models and approach to users' data have far-reaching effects. In light of concerns about the concentrations of power posed by platforms, a wave of decentralized alternatives to established platforms is emerging. As these platforms attempt to redistribute power across global communications networks, it is vital to understand how they are being built and maintained, and how values are implicated in these processes.

I approach this dissertation with a combination of approaches. I employ close technical inquiries influenced by HCI-based studies of values and design alongside broader-scoped investigations about distributed collaboration and political economy within large sociotechnical infrastructures. Similarly, because I study IndieWeb's development over multiple years, including its repurposing of existing systems, this inquiry extends to acts of repair and maintenance. This responds to calls for researchers to take a long view of technology, rather than focusing on novelty and early moments of design (e.g. Jackson 2014; Houston et al. 2016).

In situating individual design and maintenance activities within a broader sociotechnical infrastructure, this dissertation heeds Bell and Dourish's advice that "look[ing] outside of the research laboratory" can illuminate technology "as it is currently developing rather than it might be imagined to look in the

⁵Looking back, I can recognize some caveats to that experience; I was privileged to have access to a computer from a young age and parents who were supportive of my efforts to learn about Web and software development. My sense of the early Web as empowering is tempered by knowledge that this experience was not accessible to everyone.

future” (Bell and Dourish 2007, p. 139). The overall design approach employed among IndieWeb contributors embraces this perspective, and accordingly demonstrates explicit heterogeneity, messiness, and contingency. As a result, this study is able to account for complex infrastructural relationships while still leveraging insights from deep technical engagement with specific artifacts.

Methodologically, this dissertation employs a mix of qualitative and computational analyses. This combination serves to address the challenges of scale inherent to a study that encompasses both individual design activities and broader sociotechnical structures. The ability to engage in computational analyses results from the availability of logged interactional data when studying how people use digital systems, especially Web platforms that allow access to this data for research and other purposes. This approach therefore contributes to scholarship about Web platforms, where researchers are continually developing new skills to grapple with available data. This combination of approaches may be most valuable for researchers studying collaborative software development on platforms such as GitHub. Recent scholarship has highlighted that GitHub acts as a social media platform, albeit one that is quite distinctive in its focus on collaborative work (Lima, Rossi, and Musolesi 2014; Strzalkowski et al. 2019, e.g.).

Ultimately, this dissertation combines technical and material inquiries that are familiar for designers and engineers who build technology, and critical perspectives more common among social scientists. This contributes to efforts to bridge a “great divide” between these fields (e.g. G. Bowker, Star, Gasser, et al. 2014; Agre 1997a).

1.4 Structure of the Dissertation

This dissertation consists of 9 chapters. Chapter 1 has described the motivation and purpose for this study, as well as background about the role of values in the Web’s development and the IndieWeb as a site for this dissertation’s inquiry. Chapter 2 reviews theoretical perspectives that inform this study. This review is used to develop research questions and a conceptual framework, which are presented at the end of the chapter. Chapter 3 describes a research approach that combines qualitative, ethnographic methods with computational methods. Chapters 4, 5, 6, and 7 present findings as follows. Chapter 4 focuses on how values are employed in individuals’ design activities. This consists of an overview of what values are important for IndieWeb building, followed by a description of how IndieWeb standards and other features shape individuals’ design processes, and finally an account of my own experiences building software for the IndieWeb. Chapters 5 and 6 focus on the role of IndieWeb’s community. Chapter 5 presents a large-scale description of community structures, topics of concern, and correlations between activity on IndieWeb’s chat and GitHub. This serves to identify points of centralization and decentralization of influence. Chapter 6 builds on the previous chapter to examine ways in which IndieWeb’s centres of influence relate to exclusion, and how this is addressed by IndieWeb’s community. Chapter 7 describes how IndieWeb co-exists with corporate platforms. This is presented through an investigation of the development and maintenance of *Bridgy*, an IndieWeb tool for syndicating between one’s IndieWeb site and various platforms. Chapter 8 consists of a discussion that makes sense of the findings from Chapters 4-7 in the context of IndieWeb’s bigger picture. Finally, Chapter 9 summarizes the contributions of this dissertation and identifies future directions.

Chapter 2

Background and Literature Review

This chapter reviews the literature related to the overarching research question, *what kind of work is involved in building and maintaining a technical system that reimagines the social structure of the web?* In asking this question, I pay particular attention to the role of values throughout processes of design and maintenance. Further, I view IndieWeb as situated within broader sociotechnical infrastructures, and consider how this affects its design.

The first section of this chapter reviews scholarship about the relationship between technology and values. This includes key research from science and technology studies (STS) about technology and culture on a broad scale, followed by an overview of relevant research about values and design. The next section describes approaches to scholarship about infrastructures and platforms, highlighting theoretical and methodological insights that are well-suited for this dissertation's research site. Following this is a description of relevant research about political economy. This review draws attention to issues of power and control in communication systems, which are important to consider when studying how individual design activities are shaped by the sociotechnical structures in which they are situated, including political economic and cultural influences. The last area of scholarship that is reviewed focuses on labour and technological communities of practice. This section extends the preceding discussion of political economy by identifying concepts for understanding how relationships within IndieWeb's community and with broader sociotechnical structures impact possibilities for decision making while building IndieWeb systems.

The literature referenced in this review has helped me to formulate and refine research questions for this dissertation. I conclude this chapter by presenting these questions and an overall conceptual framework into which they are integrated.

2.1 Questions of values and technology

Many efforts to build alternative Web systems such as IndieWeb's demonstrate commitments to individual empowerment or other forms of social change. The belief that such change can be meaningfully pursued through technological design reflects a belief that society and technology are tightly intertwined. Furthermore, the explicit embedding of technological development in a social system—evidenced by IndieWeb's frequent in-person events and online discussion channels—reflects that this is seen as a co-creative relationship; technology and society do not determine one another, but instead shape each other.

This perspective has been a central feature of science and technology studies as well related fields such as information studies.

To situate this work in relation to dominant theories of technology, I draw on Feenberg (2002), who argues that most established theories have been of two general types. First, *instrumental theories* treat technologies as neutral tools without valuative content of their own (p. 5). This has been the most broadly accepted view of technology, and posits that technologies are essentially neutral and rational. Second, *substantive theories* argue that technology restructures the whole of social life, in the process overriding traditional and competing values (pp. 5-6). Examples include the critical theories of Ellul (1964) and Marcuse (1941). He then explains that, in spite of their obvious differences, these two types of technology similarly treat technology as an inevitability:

On the one hand, if technology is a mere instrumentality, indifferent to values, then its design is not an issue in a political debate, only the range and efficiency of its application. On the other hand, if technology is the vehicle for a culture of domination, then we are condemned either to pursue its advance toward dystopia or to regress to a more primitive way of life. In neither case can we change it: in both theories, *technology is destiny*. (Feenberg 2002, p. 8, emphasis in original)

Clearly, neither of these perspectives is suitable for this dissertation, which investigates a context in which the design of technology is explicitly political. Where entities such as corporate Web platforms are viewed as dominating, this is not regarded as a fundamental feature of technology but instead as a consequence of design decisions. In sum, IndieWeb's proposal to build an alternative to currently dominant systems rejects the notion of technology as *destiny*.

Among the most widely cited accounts of design conveying politics is Winner's (1980) essay about a series of bridges in Long Island, New York, under which one must pass to reach Jones Beach, an acclaimed public park. These bridges have particularly low clearance, which made it difficult for racial minorities and low-income groups to visit Jones Beach, since they tended to take public transit vehicles which could not pass underneath. According to Winner's account, the low clearance of these bridges was a deliberate choice "to achieve a particular social effect" (1980, p. 123), and this decision was a reflection of class and racial prejudices held by Robert Moses, New York's master builder of public works from the 1920s to 1970s. The validity of this account has been challenged (see for e.g. Joerges 1999), but this account is at least a powerful parable for understanding how objects can have political consequences as a result of design decisions (whether intentional or not).

This premise has been taken up in scholarship that aims to unpack values and design. For example, Friedman's (Friedman 1997) edited volume titled *Human values and the design of computer technology* asserts that: Software designs can have biases (Friedman and Nissenbaum 1997); the inclusion or exclusion of technical features can have ethical consequences (Tang 1997); and the categories used by information systems can have a disciplinary effect on users (Suchman 1997). These concepts have nurtured the growth of Value-Sensitive Design (VSD), a research program that works to define methods and approaches for designing technology "that accounts for human values in a principles and comprehensive manner throughout the design process" (Friedman and P. H. Kahn 2003, p. 1186). VSD advocates an iterative methodology that integrates conceptual, empirical, and technical investigations (Friedman, P. Kahn, and Borning 2002). VSD seeks to be proactive "to influence the design of technology in and throughout the design process" (Friedman, P. Kahn, and Borning 2002, p. 12). A closely related field of research is Values in Design (ViD), which "describes a research space focused on finding and naming

values challenged by emerging technologies and infrastructures” rather than proposing design strategies (Shilton, Koepfler, and Fleischmann 2013, p. 4). Studying the position of values in this space requires researchers to carefully define the term ‘values’ and its role in human behaviour, and then to articulate how and where values can be articulated in technology. I draw from VSD and ViD’s literature about defining and locating values in relation to technology, as I describe below.

2.1.1 Defining and Locating Values

Before proceeding with questions of how values are pursued through design, it is necessary to explain what is meant by values in the first place.

Shilton, Koepfler, and Fleischmann (2013) affirm that values and design literature generally uses definitions of values from anthropology, sociology, and social psychology. S. H. Schwartz (1992) summarized that across these disciplines, values are viewed as “the criteria people use to select and justify actions and to evaluate people (including the self) and events” (p. 1). This definition is congruent with conceptions about values among researchers and designers who consider values in technology, albeit somewhat vague.

In general, literature about values in design focuses on values related to “human well-being, human dignity, justice, welfare, and human rights” (Friedman and P. H. Kahn 2003, p. 1186). Friedman, P. H. Kahn, and Borning (2006) present a (non-comprehensive) list of “human values (with ethical import)” that are often implicated in systems design (p. 364). This and similar lists such as presented by Brey (2000) provide a general foundation from which to develop descriptions of values for specific design projects.

Overall, rather than adopting a top-down classification of values, VSD and related approaches generally seek to identify values that are important to relevant stakeholders. A notable consequence of this approach is that VSD lacks a firm method for distinguishing moral values from preferences (Manders-Huits 2011). This represents a departure from Schwartz’ widely known *theory of basic human values* (S. H. Schwartz 1992; S. Schwartz 2012), which frames values as stable features of human identity. S. Schwartz (2012) asserts that “values transcend specific actions and situations” and this is “distinguishes values from norms and attitudes that usually refer to specific actions, objects, or situations” (p. 4). VSD and related approaches rely on empirical observation to identify articulations of values, and therefore do not address if and how those values may extend beyond the specific actions and situations being observed. Insofar as I seek to understand the motivations, challenges, and activities experienced by IndieWeb’s contributors, I accept the limitations of an empirically based account of IndieWeb’s values. When I assert that particular actions reflect commitments to particular values, it is beyond the scope of this dissertation if and how those values extend through other aspects of IndieWeb’s contributors’ lives.

One of the pressing concerns in values and design literature is *how* values are instantiated in technology. The above definition has generally defined values as a *human* resource for making sense of the world, choosing how to act, and justifying one’s decisions. Yet research about values in a design context is largely concerned with how values are integrated into design decisions, and subsequently how they affect uses or encounters with the resulting artifact. Shilton, Koepfler, and Fleischmann (2013) identify that the notion of “values” becomes conceptually confused at the intersections of design and use:

Are values concrete attributes fundamental to individuals’ personalities and identities (Schwartz 2007)? Or are values contextual concepts based on shared negotiations of space and place (Cohen 2012; Nissenbaum 2009)? How do the values of human actors become concrete fea-

tures built into a technology (Johnson 2000; Winner 1980)? And how are values (whether fact or negotiation) mediated by use of these technologies (Jarvenpaa and Leidner 1998)? (p. 260)

In sum, the nature of values in a design context depends greatly upon how they are incorporated into, alongside, or through material artifacts.

Friedman and P. H. Kahn (2003) describe three positions for locating values in relation to technologies. First, the endogenous view posits that “designers inscribe their own intentions and values into the technology; and once developed and deployed, the resulting technology determines specific kinds of human behavior” (p. 1178). The hard version of this position is regarded as deterministic by many scholars, but a softer version has found more acceptance. The soft embodied position recognizes that technologies “do not literally embody an intention or value”, that designers themselves are shaped by external forces, and that many technologies are not widely adopted (2003, p 1179). But it still argues that those technologies that do take hold can ensure certain kinds of behaviour. Second, the exogenous position argues that societal forces significantly shape how technologies are used. This tends to be the position most evident among political economists, who emphasize the role of social power. For my study, this approach is insufficient because it fails to address ethical issues concerning technology itself. It is possible to identify many of the human stakeholders directly involved in building the IndieWeb, but the opacity of many of the systems with which IndieWeb intersects renders other human stakeholders invisible. Therefore, an approach that excludes nonhuman stakeholders is inappropriate.

The approach most often used in VSD is the interactional position, which affirms that “whereas the features or properties that people design into technologies more readily support certain values and hinder others, the technology’s actual use depends on the goals of the people interacting with it” (Friedman and P. H. Kahn 2003, p 1179). This perspective builds upon affordance theory (Gibson 1986; Norman 1999), which asserts that environments and technologies support and constrain (though do not determine) potential directions for action. Rather than denying either the endogenous or exogenous view, the interactional view recognizes “interpretive flexibility during technology design and use” (Shilton, Koepfler, and Fleischmann 2013, p. 5). In this view, technologies may be designed to support particular values, but the realization of those values depends on users’ goals, abilities, and the circumstances situations surrounding use.

Brey (2010) describes a model for understanding the instantiation of values in technology where the central uses of an artifact tend to lead to particular consequences, and those consequences tend to support the realization of particular values. For example, the central uses of a gas-engine automobile all involve driving, so tend to lead to consequences including, “gasoline is used up, greenhouse gases and other pollutants are being released, noise is being generated, and at least one person (the driver) is being moved around at high speeds” (Brey 2010, p. 44). Consequences arising from the central uses of an artifact can have the effect of either harming or supporting the realization of a value. The consequence of pollution has the effect of harming the realization of environmental values; and where cars facilitate high speed travel without reliance on public transportation, they support the realization of values of individualism and autonomy. Of course, it possible to imagine any number of alternative uses. A car could easily be left still as a storage container, a home, a planter, or a museum exhibit. In such cases, the values supported by that car could be significantly different. Even where the ‘central uses’ of an artifact are identifiable they are not set in stone.

Social constructivist studies of technology have identified a phenomenon called *closure* in which

the range of possibilities for interpretive flexibility is restricted as technologies mature. This occurs when relevant social groups are satisfied that any of their perceived problems about the technology are resolved (Pinch and Bijker 1987). Thus, it can seem that values become ‘baked in’ to technologies as the technology stabilizes. However, Humphreys (2005) observes that the stabilization that occurs during a technology’s maturation is temporary, and so possibilities for interpretive flexibility can resurface when the context surrounding a technology changes. Therefore, in addition to studying the role of values in the initial design of technologies, there is a need to address the ongoing work involved in repairing and maintaining those values.

This is particularly clear in the realm of Web technologies, which involve a multitude of explicit relationships to other systems, and since the advent of *Web 2.0* increasingly follow a model of constant iterative development (O’Reilly 2005). The ability to supplement and repurpose platforms has been one of the great sources of promise in analyses of Web 2.0 (Schäfer 2011; Shirky 2008). Nonetheless, developers who build upon platforms work with an awareness that the foundations upon which they rely may be changed or deprecated suddenly, requiring unexpected work to maintain or repair core functions. Addressing the maintenance and repair of values in technology reframes those values as “contingent and ongoing accomplishments” (Houston et al. 2016). This draws attention to variety of external pressures that can constrain values over the duration of an artifact.

A further specification about the nature of values is articulated in JafariNaimi, Nathan, and Hargraves (2015). They contend that VSD and related fields follow a two-step *identify/apply logic* in which one first identifies and understands values and then applies them to design practice. By contrast, they assert that understandings of values are developed through design practice.

...designers employ values to serve situations of practice, rather than applying them from precise definitions or following procedural steps. This finding suggests that prompts, check-lists, and step-by-step procedures fail to capture the link of values and design and certainly cannot replace the hard work of design—that is, to decide the whats and whens. (2015, pp. 99-100)

Rather than rules to be followed, JafariNaimi and colleagues treat values as hypotheses that may or may not serve specific situations. This perspective draws on Dewey’s (1891) moral theory, where he approaches ethics with a question of action: “What is the situation that demands action, and what is the action that the situation demands?” (p. 193) Does bringing our experiential understanding of a value to a complex situation help us make sense of the situation and progress towards a positive resolution? In the process of employing values as hypotheses, we not only work toward an amenable outcome, but also enrich our understandings of the values we are testing.

This is a particularly valuable perspective on values and design for studying technological development on a longitudinal and distributed basis. It is unlikely that values remain fixed across IndieWeb’s chronology or among its diverse contributors. As a result, in the course of this dissertation I expect to observe value-oriented goals being contingent upon the situations to which they are employed and the resources that accompany them. When values are used as criteria for engaging with the world, they help us make sense of situations of which they are a part. Thus, when our understanding of those situations changes, our understandings of all of their constituent parts may change, including the values we have introduced to the situation. As a result, where strategies and prioritization of designing for values vary over time, I do not view this simply as a matter of weighing these goals against each other but instead as a sign that they may be understood in qualitatively different ways.

2.2 Infrastructures and platforms

Like any networked system (and arguably all sociotechnical artifacts), IndieWeb is integrated into a broader sociotechnical architecture. IndieWeb sites and standards rely upon and extend from fundamental components of the Internet. These include protocols (e.g. TCP/IP, HTTP), language specifications (e.g. HTML, CSS), material systems (e.g. servers, networks), and other aspects of Internet infrastructure (communities, policies, regulations, and more). Furthermore, IndieWeb is largely a response against the prominence of platforms online, while also simultaneously co-opting some platform features as syndication mechanisms. Any account of the IndieWeb’s development must situate the endeavours of its community within this broader context.

Infrastructure studies and *platform studies* are materialist perspectives concerned with how technical systems underlie everyday life. This section reviews some of this literature, focusing on insights for examining the IndieWeb alongside the larger Internet as well as addressing the IndieWeb itself as a large sociotechnical system. This dissertation sits at the intersection between platform and infrastructure studies identified by Plantin and colleagues (Plantin and Powell 2016; Plantin, Lagoze, et al. 2016). Both fields share similar commitments, however there are some key differences in their approaches and the qualities of their subject matter. Infrastructure studies emphasizes “ubiquity, reliability, invisibility, gateways, and breakdown” while platform studies highlights “programmability, affordances and constraints, connection of heterogeneous actors, and accessibility of data and logic through application programming interfaces (APIs)” (Plantin, Lagoze, et al. 2016, p. 2).

Infrastructure studies is a large influence on this dissertation, especially through its conviction that studying technical details of a system can illuminate “the political, ethical, and social choices that have been made throughout its development” (G. C. Bowker, Baker, et al. 2009, p. 99). Science and technology studies (STS) has employed the concept of infrastructure to describe systems that operate invisibly in the background and tend to be taken for granted (Star 1999). Infrastructures are large and complex such as electrical grids, roadways, and telecommunication systems, encompassing both material components (bricks, wires, pipes, etc.) as well as protocols, standards, policies, environmental resources, organizations and other entities. Non-materials elements that might be designated as ‘social context’ in other fields are components of infrastructures, and therefore part of a sociotechnical system under study rather than external factors. This lens highlights the complexity of making change within large technical systems. Infrastructures are constituted by “a baffling network of relationships producing significant outcomes that no single actor seems particularly able to foresee” (Sandvig 2013, p. 89).

Studying infrastructures necessitates addressing the following features, as summarized by Sandvig (2013): “Invisibility, dependence on human practices, modularity, standardization, and momentum” (p. 96). Invisibility means that infrastructures are transparent when working smoothly but can become visible when they break down (Star 1999). Insofar as IndieWeb and similar efforts are motivated by frustration with the current structure of the Internet, and seek to address this frustration through the construction of new standards, software, and related strategies, the site of this study is founded on precisely the sort of breakdown that can bring infrastructures into focus. The process of investigating an infrastructure as a foregrounded object of study, rather than accepting it as an invisible support system in the background, has been termed “infrastructural inversion” (G. C. Bowker and Star 2000). Performing this inversion while remaining attentive to questions of values, power, and culture demands persistent attention to the human work involved in shaping technical systems. Infrastructural scholars must “guard against being distracted by the shiny material parts and uncover the tacit labor that must

always be present” (Sandvig 2013, p. 97). The modularity of infrastructures means that no single actor can easily change the whole system. Instead, the process of building and modifying infrastructures is one of negotiation. A change in one place may require adjustment elsewhere. Standardization is a critical aspect of infrastructure as it allows an infrastructure’s components to work together and facilitates interoperability with other systems. By facilitating smooth interaction, standards furthermore promote a system’s invisibility (Sandvig 2013).

Finally, momentum results from the scale, ubiquity, and modularity of infrastructures. Once formal and informal standards develop to ease interactions among an infrastructure’s components, they become difficult to change without disruption. Keeping infrastructures invisible under these circumstances — that is, avoiding disruptive breakdowns — involves managing relationships among a multitude of parts, which slows down efforts to change directions. The momentum and invisibility of complex systems increases as new versions are layered atop previous iterations (G. C. Bowker and Star 2000). As a result of this imbrication, decisions that were contentious at one time can become taken-for-granted parts of the system. Momentum is not contained within infrastructures, but also tends “to exert a soft determinism on other systems, groups, and individuals in society” (Hughes 1987, p. 55). Infrastructural momentum is a key concept for this dissertation because efforts to increase the decentralization of the Web oppose the momentum of large platforms that have come to dominate many aspects of the internet.

Studies of infrastructures tend to be concerned with how changes have unfolded over time, and how current developments relate to past decisions and conditions. In brief, “the emphasis is not on novelty but continuity and consistency with the past” (Jackson, et al., 2007). This is a significant contrast from much discourse about technology, especially in popular media where discussions about the latest and greatest new developments prevail. However, this approach demonstrates infrastructure studies’ alignment within STS, where a large body of work investigates influence between early and subsequent evolutions of technologies (e.g. Pinch and Bijker 1987; Flichy 2007). Since this dissertation examines IndieWeb using archival data spanning 2011-2019, a historically oriented view of technology is valuable for considering how IndieWeb relates to earlier and contemporary features of the Internet, and the extent to which IndieWeb’s standards and approaches may have developed momentum and influence in their own right.

Platform studies offers a related set of insights. Like infrastructure studies, platform studies is concerned with relationships between society and technical systems. Platform studies was introduced in Montfort and Bogost’s *Racing the Beam* (2009), in which they called for scholars to investigate technical details of platforms to understand their relationships to culture. Regarding the Internet, the term ‘platform’ has been used to connote various meanings about websites and apps that host user-generated content. Gillespie (2010) notes that the term implies several dimensions and works to favourably frame the legal and political responsibility of platform owners. Among the meanings of the term are:

Computational, something to build upon and innovate from; political, a place from which to speak and be heard; figurative, in that the opportunity is an abstract promise as much as a practical one; and architectural, in that YouTube is designed as an open-armed, egalitarian facilitation of expression, not an elitist gatekeeper with normative and technical restrictions. (Gillespie 2010, p. 352)

The multiple connotations of the term ‘platform’ help to bridge relationships with a variety of actors such as creators, audiences, and marketers. The intermediary position from which corporate platforms

operate has been explored in a variety of research and is a key feature of the definition of *centralized Web platforms* I presented in Section 1.2.2.

The computational dimension of platforms is especially visible in my dissertation because I concentrate on people who are building software and standards. Andreessen (Andreessen 2007) represents the perspective of programmers in his assertion that “If you can program it, then it’s a platform. If you can’t then it’s not.” Among those IndieWeb sites that syndicate to and from platforms or include other computational connections, this syndication is almost always accomplished using platform APIs. In such cases, it is important to consider how these APIs function. What types of data can be transmitted through an API, and in which directions? These questions are central to Helmond’s (2015) dissertation, in which she describes the spread of platform features throughout the Web and proposes platform infrastructure studies as a method for analyzing the “ecosystem of software platforms with digital methods” (p. 24). Where her approach “introduces the notion of platform-specific objects in addition to Web-native objects” (p. 21), many of IndieWeb’s efforts attempt to reproduce or repurpose platform features to strip them of their platform specificity.

A major difference between platforms and infrastructures is that platforms tend not to be invisible. Whereas infrastructures recede into the background, platforms’ capabilities, boundaries, and branding are visible parts of users’ experiences. When we make a phone call, we rarely think about its supporting infrastructure (unless something goes wrong); By contrast we are keenly aware of which platforms we prefer based on factors such as their affordances and social reach. For example, Google Maps’ early appeal was not oriented around the quality of its maps, but instead (1) its fluid user-interface for accessing essentially the same data that was already available through competing platforms, (2) allowing other Web developers to build upon its maps using an API (Plantin and Powell 2016). Both by hosting content created by users, and aggregating access to existing knowledge through attractive interfaces (both user-facing and programmable), platforms have become significant gatekeepers. In fact, a recent body of scholarship has focused specifically on this gatekeeping role (e.g. Gillespie 2018).

Amidst the material and rhetorical differences between infrastructures and platforms, Plantin and Powell (2016) argue that infrastructures prompt the question “Who is included/excluded - and how does this occur?” (p. 7) whereas platforms prompt a related question about how power is “decentralized and recentralized in relation to platforms” (p. 8). Attention to how power is distributed by and through platforms has been a central feature of recent platform scholarship. Notably, scholars have written about the extension of platforms into a multitude of sectors. These texts have investigated how this has served capital (Srnicsek 2017), proposed ways to build new platforms to support co-operative ownership and work (Scholz and Schneider 2016), criticized platforms’ capacity for racism (Matamoros-Fernández 2017), and challenged the suitability of platforms to support feminist politics (Singh 2018). Plantin, Lagoze, et al. (2016) assert that “the rise of digital technologies, in a neoliberal, political, and economic climate, has simultaneously facilitated a ‘platformization’ of infrastructures and an ‘infrastructuralization’ of platforms” (p. 6). Some platforms, such as Google and Facebook, have grown to a scale they describe as infrastructural and yet have been constructed according to platform logics of programmability and extracting value from data. Accordingly, they assert that platform and infrastructure perspectives can be combined to make sense of this phenomenon:

Together, they help us to see the structures, the promises, and the perils of a world where (some) platforms become infrastructures, even as (many) infrastructures are being platformized. In such a world, it has become too easy to conflate the economic logics typical

of platforms with the public interests and quasi-universal services formerly characteristic of many infrastructures. The question is not only who profits and controls, but who, and what, is cast aside along the way. (Plantin, Lagoze, et al. 2016, p. 14)

In the context of this study, it is clear that IndieWeb itself is infrastructural. It consists of standards, technical artifacts, knowledge archives and discussion forums, in-person and online events, and a large community. Most important, IndieWeb provides a foundation upon which people can build and operate personal websites with modern social features. And further, IndieWeb obviously operates within the broader infrastructure of the Web, relying and expanding upon technical standards like HTML and CSS as well as negotiating through the Web’s standardization bodies etc. IndieWeb is also a site for platforms in several respects. Most notably, there are IndieWeb-friendly platforms that provide Web content hosting and social media as a service, such as Known (<https://withknown.com/>) and Micro.blog (<https://micro.blog>). Second, many IndieWeb sites use corporate platforms as a distribution infrastructure, which explicitly gestures to an infrastructuralization of platforms. And third, IndieWeb exists as an alternative to corporate platforms, and therefore must be understood in relation to them.

Thus, questions about power, centralization and decentralization that are central to platform studies are a core concern of this dissertation. And the work involved in building IndieWeb tools involves elements of infrastructure work — the ongoing work involved in managing relationships among interoperating components, i.e. making IndieWeb’s infrastructure work toward individuals’ goals.

I propose that the question of how power is decentralized and recentralized can help address questions about how inclusion and exclusion occur. IndieWeb’s chief opponents are centralized platforms, primarily operated by for-profit corporations. The motivations for IndieWeb, as with most efforts to decentralize the Web, stem from a belief that the platform architectures deployed by large corporations serve to enhance their power at the expense of individuals. Furthermore, many of the technical artifacts created for the IndieWeb must be able to interoperate with those platforms. Even regarding aspects of the IndieWeb that do not interoperate with corporate platforms, compatibility among IndieWeb components demands coordination and consensus. Across all of these dimensions, there is potential for tensions, challenges, and contradictions among corporate, individual, and communal power. Accordingly, power relationships are significant for understanding both motivations for the IndieWeb, as well as the environments in which it is designed, maintained, and used. To address this, this study draws upon approaches from scholarship about the political economy of communication.

2.3 Political Economy

Scholars and critics of the Internet have often fallen into two camps, either celebrating the power of the Internet for coordinating movements and empowering individuals (e.g. Shirky 2008; Shirky 2010), or conversely, criticizing the Internet for encouraging shallow communication, (e.g. Lanier 2011) and echo chambers (Pariser 2012). Clearly, neither view is sufficient for studying IndieWeb, since it demonstrates simultaneous optimism for the potential of the Internet and skepticism about the reality imposed by corporate business models of centralized Web platforms. McChesney (2013) argues that political economic analysis can help move beyond the bounds of either approach to help celebrants understand “the political economic world in which they actually live” (2013, p. 16) and skeptics to develop “a better appreciation for what human agency can accomplish via technology” (2013, p. 16). For my purposes, concepts from political economy are fruitful for situating the work of IndieWeb contributors’ in (1) power

structures that operate within IndieWeb’s sociotechnical infrastructure, and (2) power structures that shape the Internet and society more broadly.

In *The political economy of communication*, Mosco (2009) presents two definitions of political economy. The first, which he calls a narrow definition, states that political economy “is the study of the social relations, particularly the power relations, that mutually constitute the production, distribution, and consumption of resources, including communication resources” (2009, p. 2). One of the limitations of this definition is that it assumes it is easy to recognize and distinguish among producers, distributors and consumers in this circulation, and these roles are increasingly blurry with regard to internet communications (Bruns 2008; Schäfer 2011; Terranova 2000). A second, broader definition states that political economy “is the study of control and survival in social life” (Mosco 2009, p. 3). The strength of this definition, according to Mosco, is that it “gives political economy the breadth to encompass at least all human activity and, arguably, all living processes” (2009, p. 3).

Mosco identifies four qualities that characterize political economy, independent of the above definitions:

1. **History.** One of the major concerns of political economy is social change. With regard to the Internet, questions have arisen about whether internet and communication technologies are creating an entirely new type of society, or whether our emerging society is an evolution of past structures. Specifically, Mosco (2009) identifies a key question about the information society: “is ours a new kind of society, as was capitalism, or is it just a form of capitalism, perhaps to be called informational capitalism?” (p. 3)
2. **The social totality.** By focusing on the social totality, political economy attempts to describe the big picture. Political economists “differ on many points but all aim to build on the unity of the political and the economic by accounting for their mutual influence and for their relationship to wider social and symbolic spheres of activity” (2009, p. 4).
3. **Moral philosophy.** Political economy is concerned with values that shape society. This drives investigations not only into how things are but also how things should be. In the case of communication, political economists generally argue for the extension of democracy and pursuit of social justice.
4. **Praxis.** Political economists challenge what they see as an artificial division between research and action. One example of this is McChesney’s (2013) co-founding of the public interest group Free Press, which advocates for democratic media policies (described on p. 93).

Across these four qualities, political economists ask, “How are power and wealth related and how are these in turn connected to cultural and social life?” (2009, p. 4).

Critical political economists are concerned not only with power relations, but also “look for moments of contradiction in social relations that offer possibilities for resistance” (Birkinbine 2015, p. 4). For example, Birkinbine (2015) studied FLOSS developers who resisted corporate involvement in their projects primarily by creating their own forks of those projects absent from corporate ownership. This was possible due to the unique legal, technical, and organizational structures of FLOSS projects, which effectively allowed developers to retain ownership of the projects of their labour. IndieWeb presents a different set of social relations and different opportunities for resisting established structures and promoting alternatives. For example, IndieWeb’s POSSE strategy attempts to redefine the relationships

between individuals and centralized platforms. Chapter 7 will investigate what is involved in this process, paying attention to power relationships among IndieWeb and platform artifacts shape possibilities for this redefinition. Even in less contentious forms of interaction, such as among various IndieWeb software, I identify constraints and opportunities that affect the ability of individuals to introduce new ideas or approaches. Ultimately, my purpose for adopting a political economic perspective is to address the questions posed at the end of Section 2.2, who is included or excluded, and how is power centralized or decentralized?

I will make use of two of the entry points Mosco (2009) identifies for political economic studies of communication: Structuration, the process of creating social relations such as those related to gender, race, and class, and commodification, where use values are transformed into exchange values.

These entry points provide footholds for analyzing the political economy of centralized platforms, which can help unpack what it means for IndieWeb to claim to provide an alternative. Furthermore, they are also useful for investigating influence and power within IndieWeb’s community.

Structuration. Structuration refers to the process of creating social relations. The concept of structuration is primarily developed by Giddens, but Mosco asserts that Giddens’ theory “tends to accentuate individual agency” and is therefore “disconnected from an understanding of power and more generally from a critical approach to society” (Mosco 2009, p. 187). To better account for power, Mosco urges that political economists consider agency and social action at both a micro level (e.g. individual-level power struggles) and broader social structure. Political economic studies tend to focus on structuration of gender, race, and class. Gender and race, especially, are important dimensions for investigating who contributes to the IndieWeb—a topic addressed in Chapter 6; like most open source software communities IndieWeb is disproportionately white, male, and educated. In addition to these well-established hierarchies, I adopt a relational perspective, focusing on how connections among people (and nonhuman actors) articulate power relationships. For example, people may relate to both corporate platforms and IndieWeb’s infrastructure as users, non-users, or developers. These roles define capacities for engaging with those structures, such as the ability to contest features with which one disagrees or repair breakdowns.

Commodification. Mosco defines commodification as “the process of turning use values into exchange values, of transforming products whose value is determined by their ability to meet individual and social needs into products whose value is set by their market price.” (2009, p. 132). In the process of this transformation, “the commodity objectifies exploitative social relations [between capital and labour] by presenting them in a congealed form that makes them seem natural” (2009, p. 131). Commodification is a useful lens through which to understand some of IndieWeb’s motivations with regard to platforms. Operators of commercial Web platforms demonstrate a double process of commodification in their extraction of monetary value from users (Cohen 2013). First, the content created and circulated by platform users is accompanied by advertisements; Second, the information surrounding this content—e.g., browsing history, retweets and likes, friend lists, demographic information—forms the basis of a market in metadata, which is sold for profit and/or used to precisely target advertisements. This process is dependent on a model of *digital sharecropping*, in which platforms provide tools for individuals to create, but retain ownership over the resulting products (at least to the extent that enables commodification) (Carr 2006). This concept has been used as one of the foremost justifications for IndieWeb’s approach, with *indieweb.org/why* prominently stating “Whatever the reason, you’re done

with sharecropping your content, your identity, and your self.”¹

With its primary goal of ‘self-empowerment’ and its opposition to the ‘corporate web,’ IndieWeb is fundamentally concerned with the balance of power among individuals, platforms, and other components of the Internet. This section has introduced foundational concepts in the political economy of communication systems. These concepts will be employed to (a) understand IndieWeb’s motivations in relation to corporate Web platforms, (b) investigate power and influence in the process of creating IndieWeb, and (c) situate IndieWeb’s political economy alongside broader social relations (i.e. the ‘social totality’).

2.4 Labour and technological communities of practice

Part of approaching my analysis from a political economy perspective is thinking about divisions of labour in the work of building and maintaining the IndieWeb. In the previous chapter, I drew from Atton (2002) to situate IndieWeb as an alternative to Centralized Web platforms. I agree with Cohen (2012) that “if we follow Atton in understanding alternative media through organizational processes and resulting social relations, we need to consider labour” (p. 208).

Labour process theory (Braverman 1998) describes divisions of labour as mechanisms through which planning and conceptual aspects of work are divided from the actual execution of work. This theory was developed to describe how, under monopoly capitalism, technologies such as assembly lines, drill presses, and sewing machines enacted a division of labour by prescribing a course and pace for work which was dictated by management. Concepts from Braverman’s approach are useful for my analysis, but it is necessary to address ways in which IndieWeb presents a dramatically different context than the factory floors about which Braverman wrote. IndieWeb contributors’ “work” is unwaged and voluntary, and there is no hierarchy of managers who prescribe labour processes onto workers. Unwaged work has often been excluded from analyses of labour, though feminist political economists have expanded definitions of labour to include unpaid work that does not create surplus value for capital but is *reproductive* in that it generates social use value (Huws 2014). It is with this understanding that I frame the activities involved in building and maintaining IndieWeb as a form of work.

In a related context, Birkinbine (2020) has asserted that alienation and exploitation can occur in voluntary contributions to Free (Libre) and Open Source Software (FLOSS), however since contributors demonstrate a sense of ownership over the resulting product, any alienation that does occur in FLOSS labour is “at least qualitatively different to more classical forms of industrial production” (p. 40). Söderberg (2008) asserts that the free and open source software “attacks alienated labour by circumventing it” (p. 10). IndieWeb’s contributors exemplify this in their commitment to build for themselves, and to embrace pluralism. These commitments will be discussed further in Chapter 4, but for now I will simply highlight that IndieWeb’s contributors are encouraged to build small projects to fit their own needs and that it is acceptable and encouraged to try new approaches to that end. And yet, it is also clear that individuals do not exhibit total autonomy over their building processes, since IndieWeb builders maintain compatibility with a complex system, build upon each other’s work, and influence each other in a variety of ways. Since I am interested in how IndieWeb’s contributors make design decisions to reflect their values, I use the concept of divisions of labour to search for cases where individual designers inherit decisions from elsewhere and to consider how this may constrain their choices.

¹In February 2019, this was edited to “Whatever the reason, you’re done with others owning your content, your identity, and your self” (<https://indieweb.org/wiki/index.php?title=why&direction=next&oldid=55419>).

Damarin’s (2013) description of a *network-organized labour process* presents several avenues for labour process control in a context—Web design—that is closely related to the IndieWeb. She draws on actor-network theory (ANT) (Latour 2005) to investigate how influence and control operate beyond (or removed from) hierarchical structures. Before discussing Damarin’s framework, I will review some core concepts of this foundation. ANT shares the conviction expressed through much of the values and technology work discussed earlier in this chapter that technologies and humans co-constitute one another. Latour asserts that sociology has been restricted by focusing only on humans, and that nonhuman actors are the ‘missing masses’ required to balance investigations of social relations (2008). To this end, he advocates a sociology where the social does not refer only to human relationships, but rather to associations among things both human and nonhuman. He acknowledges that this definition “risks diluting sociology to mean any type of aggregate from chemical bonds to legal ties, from atomic forces to corporate bodies, from physiological to political assemblies” and asserts that “this is precisely the point” (2005, p. 5). Especially in areas of study that are concerned with emerging technologies, the fast pace of innovation prevents these associations from stabilizing into a concrete relationship. Thus, these associations should be traced with an understanding of actors as mediators that “transform, translate, distort, and modify the meaning or the elements they are supposed to carry” (2005, p. 39).

Damarin employs these concepts to describe how sociotechnical networks of customers, employers, colleagues, and technical artifacts both constrain workers’ actions and offer opportunities for autonomy. Workers are embedded in this network where their “socio-technical ties exerted a sort of disciplinary pressure; in Foucault’s sense, they created demands that workers actively brought upon themselves with spontaneous, if often stoic, consent” (2013, p. 201). Three types of tie-relationships are considered. *Social ties* are relationships with human actors such as clients, employers, and colleagues, among whom Web designers must negotiate consensus in order to make decisions about their work. Given ANT’s influence on Damarin’s work, her use of the term *social* is surprising. In my analysis, I will instead use the term *human ties* to avoid confusion. *Technical ties* are relationships with technical artifacts. Computer languages, software, and technical standards may impose their own ways of working, and this is amplified by the ever present need to maintain compatibility among interoperating systems in any sort of networked system. Third, *postsocial ties* are relationships to typified others, such as a Web designer’s conception of the market or end-users. Damarin describes these typifications as similar to “Mead’s (1964) ‘generalized other,’ a standpoint individuals construct through interaction and then use to plan and judge their own acts” (2013, p. 194).

This networked perspective is a valuable lens for describing how control operates among IndieWeb contributors, users, third-party platforms, components of IndieWeb’s technical infrastructure, and post-social anticipations of future or potential users and other agents. In the remainder of this section, I discuss technological divisions of labour and communities of practice in further detail. These concepts elaborate specifically upon on influence that may be articulated through networks of ties as described by Damarin (2013).

2.4.1 Technological divisions of labour

Technical ties are of particular importance in this context. IndieWeb’s approach involves constructing an infrastructure within which to do further work. In a talk describing IndieWeb’s approach, Çelik explained, “We’re going to use the web itself to build the web we want” (Çelik 2014). Given this approach, the way in which IndieWeb’s technical infrastructure is built is important not only for assessing

its articulation of values, but also because it shapes possibilities for IndieWeb work going forward. If IndieWeb’s infrastructure constitutes technical ties alongside and through which contributors engage in work, it is helpful to consider how firmly these ties may shape labour processes. Franklin (2004) makes a distinction between *prescriptive technologies* and *holistic technologies* that is helpful to this end. Holistic technologies are those that support workers to have control over their labour process from beginning to end. With such technologies, workers may specialize in producing certain types of products, but do not specialize in specific stages of production. By contrast, prescriptive technologies are those that encourage specialization by process. Very clearly, the assembly lines and industrial labour processes at the centre of Braverman’s critique are prescriptive technologies, and as Franklin asserts, specialization by process has a long history going back at least as far as Chinese bronze casting in 1200 BCE. When examined in detail, specialization by process alludes to social organizations around work, “of discipline and planning, of organization and command” (2004, p. 15). Drawing on her expertise in metallurgy, Franklin contrasts the prescriptive technologies around Chinese bronze casting with holistic processes:

In contrast to what happens in holistic technologies, the potter who made molds in a Chinese bronze foundry had little latitude for judgement. He had to perform to narrow prescriptions. The work had to be right — or else. And what is right is laid down beforehand, by others. (2004, pp. 15-16)

Prescriptive technologies dominate contemporary society and enable us to work with great efficiency to develop increasingly sophisticated systems and products. However, this involves a trade-off: “prescriptive technologies eliminate the occasions for decision-making and judgement in general and especially for the making of principled decisions. Any goal of the technology is incorporated *a priori* in the design and is not negotiable” (2004, p. 18). In fact, beyond simply being not negotiable, the specialization by process at the core of prescriptive technologies is part of what renders infrastructures invisible—when we take infrastructures for granted in the background, we are excluded from their constituent processes and must accept them as-is.

Divisions of labour, whether articulated by human or nonhuman agents, capture planning and conceptual aspects of work by codifying them as instructions, procedures, or in the processes of technical systems. It is almost always the case that some knowledge cannot be codified and remains tacit (Johnson, Lorenz, and Lundvall 2002). Furthermore, planning is often presented as more rigid by computational systems than how it is experienced by humans. Suchman (2007) identified that computational systems (and their designers) tend to view plans as concrete sequences of steps, but in reality, people use plans as guidelines for situated action, which is continually improvised in relation to one’s environment. Therefore, divisions of labour are not deterministic because people use them to shape situated action rather than as rigid plans. In the context of the IndieWeb, technical ties may exert persistent influence upon building and maintenance, contributing to both opportunities and barriers for pursuing various design goals.

2.4.2 Learning in communities of practice

As the IndieWeb community is informally structured and is distributed geographically and institutionally, communities of practice provide a value unit of analysis for understanding the influence of various ties (especially, but not solely social ties) on learning. *Communities of practice* are based around participation in shared practices (Lave and Wenger 1991; Wenger 2011) and the learning that results thereof.

Lave and Wenger (1991) introduced communities of practice to frame learning as a social process, not simply from teacher to student (or master to apprentice) but among groups of individuals. This early work emphasized *legitimate peripheral participation*, a process by which newcomers become experienced community members by learning how to participate the activities around which the community is oriented. This is not simply a matter of learning how to perform relevant tasks, but also of learning how to be a member of the community:

The community of practice of midwifery or tailoring involves much more than the technical knowledgeable skill involved in delivering babies or producing clothes. A community of practice is a set of relations among persons, activity, and world, over time and in relation with other tangential and overlapping communities of practice. (Lave and Wenger 1991, p. 98)

Consensus in a community of practice is defined by *mutual engagement* through common activities, *joint enterprise* toward a common goal, and a *shared repertoire* of discourses and meanings (Wenger 1998).

Communities of practice are defined in large part through shared naturalization of the meaning and use of objects (G. C. Bowker and Star 2000). Lave and Wenger (1991) invoke a metaphor of ‘transparency’ to describe this naturalization, explaining that this combines “invisibility in the form of unproblematic interpretation and integration into activity, and visibility in the form of extended access to information” (p. 103.). IndieWeb’s community exemplifies this because they are formed around a shared understanding of personal websites, IndieWeb’s standards and software, corporate Web platforms, and Web development technologies. These objects are interpreted very differently by outsiders; typical platform users and platform developers each have very different understandings of those platforms than members of the IndieWeb community, who demonstrate an understanding of platforms as objects that can be contested, co-opted, or avoided. Objects that meet the local informational requirements of multiple communities of practice, while also maintaining a common identity are boundary objects that act “a means of translation” across communities (G. C. Bowker and Star 2000, p. 297). Even though the IndieWeb community, typical users of a platform, and the platform’s developers each have a different understanding of its meaning, the platform meets the needs of all these communities.

Brown and Duguid (1991) connect communities of practice to divisions of labour by describing how communities of workers collaboratively develop non-canonical practices to compensate when canonical descriptions of work (e.g. as described by management) are inadequate. They base their arguments from Orr’s studies of Xerox repair technicians, which are presented in *Talking About Machines* (1996). In this account, the technicians share knowledge largely through informal storytelling, essentially swapping war stories about challenging or otherwise interesting repairs that could not be resolved solely by following canonical knowledge passed down in manuals. Management’s efforts to direct repair processes through documentation (repair manuals) were intended to simplify labour but failed to account for the complexities of the actual work to be done.

The inadequacies of this corporation’s directive approach actually make a rep’s work more difficult to accomplish and thus perversely demands more, not fewer, improvisational skills. An ostensible downskilling and actual upskilling therefore proceed simultaneously. (Brown and Duguid 1991, p. 42)

Brown and Duguid (1991) assert that innovation and organizational change are often a result of non-canonical practices that challenge established or conventional ways of working. A consequence of this

is that organizations come into conflict with their members if they fail to recognize the value of this contribution.

Between Braverman's (1974) pessimistic view and Adler's (1987) optimistic one, lies Barley's (1988) complex argument, pointing out that the intent to downskill does not necessarily lead to downskilling (as Orr's reps show). But the intent to downskill may first drive noncanonical practice and communities yet further underground so that the insights gained through work are more completely hidden from the organization as a whole. (Brown and Duguid 1991, p. 53)

Non-canonical practice is a fruitful concept for understanding relationships between IndieWeb builders and third parties. This is evident in Chapter 7, where I discuss a case in which IndieWeb builders use platform APIs in ways that fall outside platform operators' canonical expectations of what those APIs are for.

As pointed out by Cox (2005), Brown and Duguid's account does little to address power relations within a community of practice. Community of practice theory in general has been criticized for not addressing internal hierarchies (B. Davies 2005). Even where communities of practice theory recognizes power relationships within such communities (e.g. Lave and Wenger 1991, recognizes differences among masters and apprentices, experts and newcomers), it does not tend to analyze internal power struggles (Fox 2000). As Fox (2000) asserts, ANT provides a framework for such analysis. Accordingly, I incorporate concepts from communities of practice theory as tools for studying particular relationships using Damarin's (Damarin 2013) framework for studying labour process control in networked systems.

2.5 Theoretical approach and Research Questions

This chapter has presented relevant literature about values and technology, including efforts to integrate and understand the role of values in design as well as theoretical approaches for investigating systems such as infrastructures and platforms. Through this review, I have identified concepts for understanding how design activities conducted by individuals or small groups are embedded in broader communal and infrastructural contexts. Based this literature, I present research questions for investigating the activities involved in building the IndieWeb while attending to power relations across this scale. As I present these questions, I weave them into a conceptual framework.

I begin with two questions about the definition of values, and how they are employed in design activities:

- **RQ1:** What values are central to the IndieWeb, and how are they defined, selected, and critiqued?
- **RQ2:** How are values employed in design activities?

I consider these questions as part of a whole, where the process of employing values in design activities feeds back upon how those values are understood. Following the interactional perspective most common in values and design research, I consider the expression of values in designed artifacts to be contingent upon the context of interactions. I further accept the notion of values as hypotheses (JafariNaimi, Nathan, and Hargraves 2015), which highlights that individual and communal understandings of values are contingent upon their relationship to the situations in which they are employed. Thus, processes

of employing values in design are examples of processes through which values are defined, selected, and critiqued.

Of particular importance here is that breakdowns can occur in the tools upon which designers rely, including third-party platforms and infrastructures. Breakdowns may occur as designers are building or after the fact, for instance a platform upon which designers relied is shut down or modified. Breakdowns not only cause a practical problem, but are also opportunities for infrastructural inversion, in which they are brought into the foreground as a result of failing to function invisibly in the background (Star 1999). When investigating how values are employed in design activities, I include responses to breakdowns. Thus, I include maintenance in my conception of “design activities,” and therein acknowledge that feedback between employing and defining values may occur over an extended period of time. Ultimately, this perspective echoes Suchman’s (2007) analysis of plans and situated action. Plans extending from values and other beliefs may be composed in advance, but each step into action can change one’s situation, thus demanding that the plan and its constituent beliefs be continually considered and potentially adapted.

Finally, RQ3 turns to questions of power, drawing upon the political economy and labour scholarship referenced in this chapter:

- **RQ3:** Is power over planning and executing design activities centralized or decentralized among stakeholders, and is anyone included or excluded as a result?

Scholarship about the intersection of infrastructures and platforms (Plantin and Powell 2016; Plantin, Lagoze, et al. 2016) has identified inclusion/exclusion and centralization/decentralization of power as important considerations of these structures, respectively. I have asserted that IndieWeb sits at this intersection, and thus consider how inclusion/exclusion and centralization/decentralization may operate in tandem. It is through this question that my research extends beyond the activities and beliefs of individuals, to also consider influences from within IndieWeb’s community as well as from external infrastructures.

Since I am concerned with the work involved in building the IndieWeb, I have focused my attention on divisions of labour as a mechanism of control over that work. Given the informal setting in which IndieWeb building takes place, I adopt Damarin’s (Damarin 2013) framework for studying labour process control in networked systems. Damarin classifies three types of ties through which influence flows: social ties, technical ties, and postsocial ties. I adjust her language slightly to reflect my agreement with Latour (Latour 2005) that ‘social’ can refer to relationships with both humans and nonhumans. Therefore, I consider influence among:

- Human ties
- Technical ties
- Postsocial ties

With regard to technical ties I consider the strength of labour process control using Franklin’s (2004) classification of prescriptive and holistic technologies, as described earlier in this chapter. Specifically, prescriptive technologies demand specialization by process, and therefore restrict the ability to make decisions beyond one’s area of specialization. Thus, when investigating IndieWeb contributors’ relationships with technical ties, I will consider the extent to which those ties may constrain types of planning or decision making.

When investigating human ties, I focus on interactions among members of IndieWeb’s community. Whereas the definition and employment of values and hypotheses can occur at a scale of individual designers, this is not sufficient for recognizing questions of power and influence. To address influence among human ties, I view IndieWeb as a community of practice oriented around a shared endeavour of building websites and related infrastructures to support a common goal. Communities of practice (Lave and Wenger 1991) highlight the relationship between knowledge and membership in a community. Specific forms of practice are recognized as legitimate, and this legitimation is instrumental for bridging individuals’ beliefs and practices to a communal scale. This perspective provides a means of investigating how particular understandings of values and technologies may be naturalized across IndieWeb’s community, and how this may project influence throughout.

Postsocial ties are relationships to typified others. Rather than yielding direct influence, these ties are representations of actors who are “personally absent” (Damarin 2013, p. 194) such as anticipated end-users or potential clients in the future. Unlike the designers in Damarin’s study, IndieWeb’s contributors need not meet the needs of clients, employers, or their postsocial equivalents in order to maintain their livelihood and so will likely experience these levers of control as less forceful. Nonetheless, my analysis will consider ways in which IndieWeb’s contributors may structure their work in accordance with beliefs about abstract ‘end-users’ or other postsocial ties.

In sum, this approach describes a way for control to operate in contexts that lack a clear management-worker division of labour. I consider how technical, human, and post-social ties may influence individual IndieWeb contributors’ work. The phenomena referenced in the three research questions do not proceed linearly, but rather in tandem. The definition of values (RQ1) and employment of values in design activities (RQ2) operate as a cycle. While those working on IndieWeb projects may (at least sometimes) experience this cycle at an individual scale, RQ3 examines how power over planning and executing design decisions may be centralized or decentralized across IndieWeb’s community and through interactions with broader systems.

Chapter 3

Research Design

This chapter presents an overview of the dissertation’s research design. The methods used in this study reflect the commitments and concerns that shaped its questions. The core goal of this project is to investigate the work involved in building the IndieWeb, with particular attention to the relationship between values and building activities.

Like E. G. Coleman (2010), I see my task as related to *provincializing* digital media by investigating “how, where, and why” it culturally matters (p. 489). This means paying attention to local forms of engagement with digital technologies, particularly with the unseen work involved in building, maintaining, using, and contesting them. This works to counteract popular concepts of technology (or knowledge of any kind) as universal or somehow outside of space. For example, D. Miller and Slater (2003) studied the Internet as grounded in a specific context (Trinidad), arguing that this approach offered more value than previous studies that generalized ‘cyberspace’, ‘the internet’, or ‘virtuality.’ They argue “the Internet as a meaningful phenomenon only exists in particular places” (D. Miller and Slater 2003, p. 1) and so focusing on one particular site is far from a limitation.

This view, in which local particularities are meaningful features of a study rather than an impediment for generalizability, is one I share. In this sense, I view IndieWeb as what Luker (2008) calls a *data outcropping*, a site in which the phenomenon I am studying is especially visible. This general phenomenon is the set of activities and obstacles involved in building and maintaining Internet systems that reflect their creators’ values. IndieWeb presents a valuable case because its contributors exhibit the following characteristics: (1) They are explicitly engaged in building internet systems, and they explicitly connect that activity with the achievement of values related to individual autonomy; (2) They demonstrate a nuanced understanding of Web technologies and an uncommon capacity for altering their functions; (3) IndieWeb has operated as a coherent community since 2011, and thus demonstrate not only design and building, but also the activities involved in *maintaining* a large system. Accordingly, IndieWeb provides a basis for generalizing, as Luker (2008) puts it, “not representative of the larger *population*, but of the larger *phenomenon*” (p. 103). Studying IndieWeb does not provide a basis to proclaim generalities about all other decentralized Web projects, alternative social media, or otherwise value-driven efforts to change the Web. But it does provide an opportunity to develop theory about the sorts of activities, obstacles, and other factors involved in this endeavour.

Because I am largely focused on the work of building the IndieWeb, Randall et al.’s discussion of ethnomethodologically informed ethnography also informs my approach:

Ethnomethodologically informed ethnography seeks to understand the organization of work, its flow, and the division of labor from the point of view of those involved in the work. Because work settings are organized around, through, and within a division of labor, work activities are necessarily seen as interdependent. Understanding how people coordinate their work in realtime, moment-by-moment, how they orient to the ‘working division of labor’ to make sense of what they are doing, is a feature of ethnographic explication. (Randall, Harper, and Rouncefield 2007, p. 121)

Accounting for these activities while bearing in mind my overarching research goals means acknowledging the forces through which divisions of labour are exercised. This introduces methodological challenges of scale. Some of the design activities related to IndieWeb can operate at a personal scale, especially given IndieWeb’s call for people to build for themselves, and yet individual projects still take place in a larger community and integrate with both IndieWeb’s technical infrastructure and larger Internet systems. These questions of scale become especially important with regard to longer-term maintenance because components in this system are updated, changed, or removed as time passes.

To address these areas, this study incorporates multiple methods. Namely, conventional ethnographic methods such as participant-observation, document analysis, and interviews; critical making (Ratto 2011) as a vehicle for auto-ethnographic analysis; and both qualitative and computational analyses of archived interactional data such as chat logs and software development logs. In the same sense that I view IndieWeb as a data outcropping of a broader phenomenon, I have selected specific sites of my data collection as data outcroppings suitable for understanding the activities and power relationships that shape IndieWeb’s construction and maintenance across these scales.

In this chapter, I discuss IndieWeb as a research setting, identifying the sites and activities through which IndieWeb’s community builds together. Following this, I describe two data collection and analysis approaches that are combined in this study, participant-observation (including interviews) and computational analysis of logged interactional data. With regard to the computational analyses, I combine topic modelling and network analysis, and explain how results from each approach are analyzed in combination. After having described the individual methods used in this dissertation, I discuss how they are integrated to study IndieWeb across multiple scales. Finally, I discuss ethical considerations related to this dissertation and identify limitations.

3.1 Research setting

The IndieWeb community is organized through both online and in-person activities. I will briefly summarize these activities and then describe how they have been incorporated into this project’s data collection. Community is a major part of the IndieWeb, and IndieWeb’s community is formed across several types of sites, such as:

- **Events.** IndieWeb began as an event called IndieWebCamp in Portland, Oregon, USA in 2011. Between then and 2020, 60+ IndieWebCamps have been held in the United States, United Kingdom, Germany, The Netherlands, and Istanbul, with three online IndieWebCamps held in 2014, 2019, and 2020. IndieWebCamps are two-day events, typically held over a weekend. The first day begins with an introductory talk, and larger events often include invited talks by keynote speakers. After the talks, attendees co-organize sessions in which to brainstorm and discuss topics of interest.

These sessions are not scheduled in advance, but instead anyone may propose sessions, which are then schedule according to consensus.¹ During the second day of an IndieWebCamp, attendees collaborate or independently build software, websites, or designs, which are often (though not always) related to discussions from the previous day. Since 2016, IndieWebCamp Portland was renamed to IndieWeb Summit.

Additionally, several cities host regular (usually biweekly) events called *Homebrew Website Club*, named after the famous *Homebrew Computer Club*. These are one-hour meetups in which participants work on their website or other small projects. Most Homebrew Website Club also include an optional quiet writing hour before the event proper. In addition, virtual Homebrew Website Clubs have been organized for those who don't have an in-person event near them.

- **Online Chat.** Online chat is the predominant medium for day to day conversations among IndieWeb's community. IndieWeb hosts several channels on Freenode, an Internet Relay Chat (IRC) server. As of February 2020, IndieWeb chat channels include:
 - **indieweb:** For general IndieWeb discussions.
 - **indieweb-dev:** For development focused discussions.
 - **indieweb-meta:** Discussions about IndieWeb's community and infrastructure.
 - **indieweb-wordpress:** Discussions about IndieWeb development for WordPress (a popular content management system for websites).
 - **indieweb-chat:** Off topic conversations.
 - **microformats:** Discussions about Microformats, a data format for representing common entities (people, events, blog posts, etc.) in HTML. Microformats are one of the major building blocks of the IndieWeb.
 - **knownchat:** Discussion about Known, an IndieWeb-friendly content management system.

With the exception of *indieweb-chat*, these chat channels are publicly archived. Archives are accessible at <https://chat.indieweb.org>.

- **IndieWeb.org Wiki.** IndieWeb.org runs on Wiki software, which means that the site can be edited by anyone² and contains a history of every edit. The wiki is used to document many features of the IndieWeb, such as definitions of what IndieWeb is, its motivations and principles, descriptions of standards and IndieWeb-related projects, pages for IndieWeb events, and profiles of IndieWeb community members. Additionally, the wiki includes articles about other Web technologies, which are typically framed in relation to IndieWeb (i.e. how they differ from an IndieWeb approach, how they could be integrated into IndieWeb workflows etc.). Where online chat is the main medium for day-to-day discussions, IndieWeb's wiki is its main medium for longer term archives.
- **Personal websites.** Since IndieWeb is devoted to building personal websites that can communicate with each other, these comprise an important part of IndieWeb's community. There is no

¹IndieWebCamps follow a BarCamp “unconference” format (and IndieWeb co-founder Tantek Çelik is also one of the co-organizers of the first BarCamp event)

²Although anyone can make changes to a Wiki, IndieWeb.org is a special case because it uses a custom log-in mechanism called IndieAuth. IndieAuth allows people to log-in by validating their control of a personal domain name (similar to using *Facebook Connect* to log into a third-party site using one's Facebook credentials). This de facto limits access to editing IndieWeb's wiki to people who have set up this IndieWeb technology on their personal website.

exact measure of the number of IndieWeb sites and they (by design) range significantly in structure, content, and appearance. Since IndieWeb encourages using one’s personal site as the centre of one’s online experience, these are a valuable resource.

- **Technical specifications.** IndieWeb specifications describe protocols for communicating between two entities (such as between two Web servers) and standardized formats for storing and presenting data on the Web. Standard protocols and formats provide a basis for different IndieWeb projects and websites to interoperate, which is a feature part of IndieWeb’s design. Several of these specifications have been published by the World Wide Web Consortium (W3C), the main standards body for the Web, founded by Tim Berners-Lee. Some IndieWeb specifications have not been published by the W3C so are instead presented at <https://spec.indieweb.org>.
- **Project repositories.** Most IndieWeb applications, code libraries, and related tools are hosted on GitHub, a platform for software code and other version-controlled repositories. GitHub repositories include a detailed history of revisions to source code and other documents, making it possible to identify how a project has changed over time. In addition, bug-reports, feature requests, and other discussions are posted as “issues.”

3.2 Participant-observation

IndieWeb’s activities are heavily documented, and it is possible to access online archives such as chat logs, wiki revisions, software code, blog posts, and meeting minutes dating from IndieWeb’s beginning in 2011 to the present. The first stage of data collection for this project was to review these archives. During this early stage I was more observer than participant, and I focused on “mapping the terrain” (Luker 2008, p. 168) to understand formal and informal power structures in IndieWeb’s community as well as get a sense of community norms, values, and practices.

Since IndieWeb’s chat is essentially its central medium for day to day discussions, it served as a useful centre from which to launch other archival inquiries. Discussions in chat are often contextualized by links to related blog posts, IndieWeb wiki articles, GitHub repositories, and other documents. As a result, it was possible to treat the chat archives as starting place for much analysis, following links to other resources that arise in chat discussions.

In one of the earliest analyses in this study, I downloaded IndieWeb’s chat archives. The archives begin in 2011 and include hundreds of thousands of messages. Since I was foremost interested in discussions pertaining to values, I filtered these archives using keyword searches for several terms: “value”, “principle”, “ethos”, “ethic”. Messages containing the term “value” and variations (“values,” “valuing”) were not particularly useful because the term “value” has multiple meanings—it may refer to ethical values, as I’m concerned with, but can also be a synonym for “worth” or refer to the value (i.e. content) of a variable or property in a computing system. The term “principle” returned the most relevant results because IndieWeb’s wiki includes a frequently cited page describing “principles for building on the indie web” (IndieWeb.org 2020i). One of the motivations of this inquiry was to identify disagreements and debates around values, so I paid particular attention to conversations in which participants questioned the rationale for particular decisions, defended their own choices, or explained why IndieWeb does things a certain way (e.g. to newcomers). I copied exemplary passages into field notes and recorded my thoughts on each.

In June 2017, I attended my first IndieWeb event, an annual IndieWeb Summit (IWS) in Portland. At this time, I was still in the process of preparing a research proposal, so this was more a general fact-finding mission than formal data collection. After this preliminary step, I began active participant-observation in Fall of 2017. First, I added IndieWeb features to my personal website, which involved reviewing IndieWeb documentation, asking for help to troubleshoot problems (and reporting bugs), and documenting my process. Second, motivated by one of IndieWeb’s principles to “build for yourself” (IndieWeb.org 2020i)—i.e. build something that satisfies your own needs rather than attempting to build for others’ hypothetical needs—I reflected on my knowledge of what IndieWeb tools existed and what I felt was missing. At the IWS I attended, it seemed like most people were active publishers of content on their websites and had focused on building tools for posting various types of articles, notes, and replies, as well as several tools for self-tracking (e.g., food intake, travel, location-histories). Both because I am relatively reserved when it comes to publishing frequent updates online, and most importantly because this dissertation presented a need for me to keep up with what other people are posting to their IndieWeb sites, I decided to build a feed-reading software. The resulting software can be used to follow IndieWeb (and other) websites, organizing their posts into a social timeline similar to a Twitter or Facebook feed.

IndieWeb’s architecture emphasizes interoperability among many individual components. As a consequence, my efforts to build could not be performed in isolation. Even though I was working on relatively small projects, getting them to actually work necessitated coordination. Some of the technical specifications upon which I relied were still drafts and included gaps that only became evident as bugs in software that implement those specifications. Identifying and addressing these problems involved discussions with people involved at various levels of IndieWeb’s development: specification editors, developers of other applications, and users who submitted bug reports. Building, therefore, was an effective step towards understanding social relations among IndieWeb’s community and infrastructure. The processes of building this software is discussed in Chapter 4, alongside other descriptions of building activities.

As well as building software and participating in online discussions, I attended additional IndieWeb events where feasible. I attended two additional IndieWeb Summits in-person (2018 and 2019, both in Portland). I also remote-attended other IndieWebCamps and several Virtual Homebrew Website Clubs, listed in Table 3.1. My participation at these events involved both discussions and working on IndieWeb-related projects. I both worked on my own projects, which involved soliciting guidance and feedback from other event attendees, and assisted other attendees with their work. Lastly, especially during in-person events, participation included attending social events such as meals.

Through participation and observation at these sites, I have investigated the activities involved in defining and adopting values and technical approaches, building systems that work together, and keeping them running.

3.2.1 Critical making

Making was an especially important form of participation during this study. One of the founding principles that motivated the founding of IndieWeb was a commitment to be “creators instead of talkers” (IndieWeb.org 2015). As a result, actively contributing to the development of IndieWeb tools is necessary to establish my legitimacy as a participant. More importantly, building the IndieWeb is the phenomenon under study in this dissertation, and engaging in building activities contributed to insights about the processes in which IndieWeb contributors are engaged. To these ends, my participant-observation dur-

Table 3.1: IndieWeb events attended during my research.

Date	Event	Attendance
June 2017	IndieWeb Summit, Portland, OR	In-person
June 2018	IndieWeb Summit, Portland, OR	In-person
Sep 2018	IndieWebCamp, New York, NY	Remote
Feb 2019	IndieWebCamp, Austin, TX	Remote
March 2019	IndieWebCamp, New Haven, CT	Remote
June 2019	IndieWeb Summit, Portland, OR	In-person
Feb 2020	IndieWebCamp Online	Remote
Various dates	Virtual Homebrew Website Club	Remote

ing this study includes working on building software and participating in other activities related to IndieWeb’s technical implementation.

In doing so, I align my work with *critical making*. Ratto (2011) introduces critical making as including three stages that are functionally intertwined and may proceed in any order (most likely with some degree of simultaneity):

1. Literature review and compilation of useful concepts and theories.
2. Collaborative design and building of technical prototypes as a means of exploring theoretical concepts.
3. Iterative processes of reconfiguration, conversation, and reflection. This involved “wrestling with the technical prototypes, exploring the various configurations and alternative possibilities, and using them to express, critique, and extend relevant concepts, theories, and models.” (2011, p. 253)

Throughout these stages, critical making does not emphasize the end-product, but instead finds value in “the act of shared construction itself as an activity and a site for enhancing and extending conceptual understandings of critical sociotechnical issues” (2011, p. 254). At the core of this endeavour is an effort to bridge a ‘great divide’ between technical and social disciplines in the academy (see G. Bowker, Star, Turner, et al. 1997). Specifically, Ratto’s (forthcoming) description of challenges when combining technical work with critical scholarship. Agre (1997b) describes how technical disciplines encourage instrumental understandings in which technologies are evaluated according to the efficiency or effectiveness of their performance, and that this conflicts with critical modes of thinking that involve recognizing the contingency of taken for granted values and assumptions.

Ratto (2011) proposes that “conceptual but deterministic social theories treat technologies as ‘matters of fact’ and therefore as a backdrop to the indeterminacy of social action” (p. 259). This leverages Latour’s (2004) distinction between *matters of fact* and *matters of concern*. Matters of fact are immutable and robust, whereas matters of concern are recognized as contestable and potentially worth investing time and energy to influence. Ratto’s argument echoes earlier claims about media, such as

Dewey’s proposal that the majority of people consume mass-media without being able to observe the processes and apparatuses through which it is produced:

Not understanding its “how,” they cannot use and control its manifestations. They undergo the consequences, they are affected by them. They cannot manage them, though some are fortunate enough—what is commonly called good fortune—to be able to exploit some phase of the process for their own personal profit. (Dewey 1927, p. 165)

One of the celebrated features of the corporate Web platforms to which IndieWeb presents an alternative is that they do engage a broad assortment of people in *producing*, not merely consuming, media. However, what remains true is that many of the apparatuses through which social media is distributed, sorted, and filtered are opaque to most people (see for e.g. Gillespie 2018; Cheney-Lippold 2011). By engaging in debate and implementation of Web standards and software, IndieWeb’s contributors demonstrate an awareness of these technical objects as matters of concern. And similarly, insofar as IndieWeb includes tools that repurpose platform APIs as distribution infrastructure,³ they challenge the immutability of those structures as well. As a result, my participant-observation emphasizes building as a primary mode of participation.

While engaged in building, there is a risk of naturalizing assumptions of instrumentalism that underly much technical work. One of the main strategies employed in critical making to avoid this is to focus on process over an end-product. My first exposure to critical making was in a pedagogical context,⁴ and here process was clearly precedent over product. Assignments in the course generally proceeded as follows. First, we read and discussed scholarship concerning some relationship between technology and society, such as varying definitions of morality in technology, or relationships between technology and space. Then, in groups, we designed and built an Arduino-based technology that articulated or explored theoretical concepts from our readings. Then, both by presented our projects to the class and by writing course papers, we reflected on the process of building our object, identified material and conceptual challenges, and explained our success (or failure) to incorporate theory into making.

During the material parts of this work, I (and others) struggled to avoid slipping into instrumental ways of thinking. Specifically, making technology presents a series of technical problems to be solved, and in the course of solving those problems—*i.e.* *trying to get the thing to just work!*—it is difficult to remain reflexive about why one makes certain decisions and not others. Every designer or engineer probably has stories of getting sucked into an interesting technical problem, investing hours into it, and only later realizing that another approach would better reflect their goals or be simpler. In fact, I would argue that such experiences suggest a *flow state* (Csikszentmihalyi 2009), a feeling of full absorption in an activity that has been regarded as a positive trait in many fields, including software development (e.g. Kuusinen et al. 2016). To aid reflection in spite of this tendency, a large part of the critical making course centered around documenting our processes. We documented our conversations, our design decisions, our thinking, the problems we encountered, and attempts to resolve those problems both successful and unsuccessful. We took notes, photos, and drew sketches. And no matter how detailed our accounts were, we still found ourselves struggling to remember how or why we implemented some aspects of our project. Though never perfect, this effort to document everything and to engage in constant and iterative reflection helped us to at least reduce the impulse toward instrumental, outcome-oriented thinking.

³These include a practice called *Publish on Own Site, Syndicate Elsewhere* (POSSE), described in detail in Chapter 7.

⁴In the second year of my PhD, I took Ratto’s Critical Making course as an elective, and later became a teaching assistant for the same course.

The making projects I undertake during this dissertation do not precisely match the experience I had during my critical making course, nor those described in Ratto (2011), where he presents an overview of critical making in the context of workshops. Critical making as a classroom or workshop activity can focus on projects that need not (and probably won't) be used in the future. Rather than building something useful, some of the most effective critical making exercises attempt to operate at the most interesting intersection with theory. By contrast, I was making things that I really wanted to use, which is a necessary part of building things the IndieWeb way. However, ultimately this did not detract from my efforts to document and reflect on making processes. I took detailed notes during each stage of work, paying particular attention to how I attempted to articulate my values through design, and how material constraints affected what I perceived as possible or feasible.

3.2.2 Interviews

Interviews were conducted with 15 IndieWeb contributors. Interviews were semi-structured, following the protocol outlined in Appendix C. Interviews ranged in length from 50 minutes to 3.5 hours. Almost all interviews were conducted online via voice or video conferencing. The exception was one interview which was split into two sections; the first half of the interview was conducted online and the second half was conducted in person immediately before an IndieWeb event.

The interviews had several functions. Interviews were useful for investigating motivations, goals, relationships, and design decisions that were not as evident in other aspects of this study. One section of the interview was concerned with identifying participants' prioritization of different goals as well as the extent to which individuals' goals varied from the broader community. Another section of the interview focused on identifying sources of disagreements, as well as how they were resolved. Additionally, participants were asked how their day jobs and other aspects of their daily lives related to the IndieWeb, such as whether their day jobs overlapped with IndieWeb, and what sort of obligations might limit the time they could spend on IndieWeb projects. Finally, one of the most fruitful subjects that came up during interviews was about diversity and inclusion in the IndieWeb.

Interview participants were recruited by direct messages over email, direct messages over IRC or Slack, and Twitter. In my dissertation proposal, I began with a goal of interviewing 20 individuals. During data collection this was scaled back to a goal of 10-20 interviews. Recruitment occurred on a rolling basis, so it was possible to adjust recruitment strategies based on findings (such as suggestions about whom to interview) from earlier interviews. Recruitment was conducted with a plan to stop scheduling interviews once their content became repetitive—i.e. once the interview data became saturated.

In attempting to achieve theoretical saturation, I selected interview subjects that are representative of the phenomena described by my study (Luker 2008). All interview participants must have contributed to IndieWeb projects. This could consist of direct contributions to software or standards, substantial customizing of their own IndieWeb site (defined as a site that used Microformats and Webmentions), or ongoing participation in IndieWeb's chat discussions or wiki. An effort was made to interview individuals at both the core and periphery of IndieWeb. I defined core contributors as those who have made significant contributions and who are likely to have an above-average influence on the direction of the IndieWeb. These are people who are especially visible on IndieWeb chat and who have made substantial technical contributions to multiple IndieWeb tools. I also endeavoured to recruit individuals who have contributed to IndieWeb in some way (even if primarily by working on their own site), but who appear to be less central than the first group. In practice, it was difficult to clearly delineate among these

Table 3.2: Basic demographics of interview invitations and participants⁵

	All	Male	Female	White	Not white
Invited	21	14	6	19	2
Interview completed	15	12	3	13	2

groups because there are many types of participation to compare. Therefore, I took the notion of *core/periphery* as a guideline to ensure I interviewed people who reflected varying positions among IndieWeb’s community.

It was difficult to recruit from less active participants. For example, I tried to recruit several individuals who had contributed to IndieWeb in the past but seemed to have moved on, and none of these people responded to my interview requests. By contrast, interview recruitment among the most active participants was extremely easy.

Finally, in attempting to reflect a diverse range of experiences related to IndieWeb, I attempted to find interview participants representing a range of cultural backgrounds and genders. As evidenced in Table 3.2, this was a challenge, since a significant portion of IndieWeb’s contributors are white, male, and based in North America or other Western regions. This is a common trait among online tech communities (Nafus 2012; Kuechler, Gilbertson, and Jensen 2012; Reagle 2013; Vasilescu, Capiluppi, and Serebrenik 2012). Pursuing a more culturally diverse set of interview participants could have added a larger variety of experiences, and particularly aided in connecting IndieWeb to broader structural issues related to race, gender, and class. However, for the purpose of addressing my research questions, focusing on cultural diversity to that extent could misrepresent the reality of working on projects like IndieWeb.

Interview demographics were collected through a series of questions at the end of the interview. I prefaced these questions with a reminder that participants could abstain from answering if they preferred, in case they might be reluctant to share their age, gender, race, education level, or place of residence. I based these on demographic questions from Statistics Canada’s 2018 Labour Force Survey (Statistics Canada 2018), modifying their phrasing to fit a conversational interview rather than a written survey.

At the beginning of the interview, participants were asked how they would like to be identified in this dissertation and related publications. Almost all participants indicated they were willing to be referred to by their real name, with a few participants indicating they would like to think about it or decided on a case-by-case basis. In most cases, there is no analytical need to refer to interview participants by their real name. However, in some cases the content of a quotation could reveal a participant’s identity even if his or her names was masked. For example, during several interviews, participants discuss specific projects they have worked on, in which case it would be simple to look up those projects and identify them. Further, beyond being *willing* to be attributed by name, it is also possible that that interview participants may prefer attribution as a means of acknowledging their contribution. Since this was unclear in many participants’ initial responses, I sent a follow-up question to confirm if they *preferred*

⁵Since individuals who were invited but did not participate in an interview had no opportunity to self-identify their gender and race, demographic information in this table is based on my observation at the time of recruitment. Chapter 6 includes a more detailed account of interview participant demographics based on their self-identification.

such attribution. As a consequence, I default to attributing interview quotations anonymously, with the following exceptions:

- A participant indicated that they prefer to be identified by name.
- A participant indicated that they are willing to be identified by name and the content of a quotation could identify the speaker even if unattributed.

Finally, participants were also asked if they wish to verify quotations before they are published, and this gave them an opportunity to rectify any misrepresentations of their words.

3.2.3 Coding and Reflecting on Field Notes

I wrote 203 field notes documents between November 2016 and February 2020. These included the following: Reflections about interviews that identified key concepts and patterns; excerpts and notes about passages from IndieWeb’s chat logs; notes about my processes of building IndieWeb software (this includes many excerpts of source code as well as verbose error messages that are sometimes over a page long); notes about IndieWeb events I attended either in-person or virtually; and other reflections about my participant-observation. Documents range in length widely, from a 29-page collection of notes about one month’s worth of software development, to less than a page of commentary on a day’s chat logs. In total, these field notes consisted of 53,932 words. Field notes and interview transcripts were coded in *Atlas.TI* using an *initial coding* process (Saldaña 2013, p.100). Initial coding has been referred to as “open coding” in early grounded theory literature. Following Charmaz (2006), Saldaña uses the term “initial coding” to indicate that this is only the first cycle of a longer process. This first step consists of quickly proceeding through documents and coding sections with eclectic codes. Given my interest in this dissertation with how participants attempt to instantiate their values through design activities, initial coding tended to emphasize values and goals (e.g. autonomy, diversity and inclusion), actions and design strategies (e.g. modular design, building for oneself, outreach to other communities), and obstacles (e.g. time constraints, difficulty interoperating with third party technologies, lack of knowledge about certain issues). However, I also remained open to other patterns that might emerge, and coded any concept that seemed relevant, especially concepts that occurred frequently throughout multiple interviews.

Initial coding resulted in a large body of codes, and the next step was to reduce this into a more manageable set of concepts. I merged overlapping codes and removed those that occurred infrequently or failed to address my research questions. Through this process, I identified significant themes, focusing on interplays of influence among individual, communal, and infrastructural structures and how these shaped decision-making and action. I also identified how concepts established during coding related with one another to describe tensions among goals, activities, and various structures, and approaches for resolving those tensions.

Through this process, I viewed coding as a tool for orienting my thinking in relation to the data and my research questions. One of the significant challenges of this dissertation was integrating concepts across multiple scales. As will be discussed presently, methods employed to analyze logged interactional data were significantly different from the field approaches described so far. These analyses were not subject the same or commensurable coding processes. As a result, while coding served to identify important concepts and their relationships to each other, writing itself served as an important analytic tool insofar as it provided process through which to bridge concepts developed from participant-observation and those established from broader-scale computational analyses.

3.3 Logged interactional data

Studies of online communities have often incorporated conventional ethnographic methods alongside novel forms of analysis suited to the nature of digital data. Online conversations and other interactions are typically archived, and these archives provide opportunities to study “traces” of past activities.

Geiger and Ribes (2011) introduce *trace ethnography* as an approach for studying such traces to “capture many distributed phenomena that are otherwise difficult to study” (p. 1). They illustrate this approach with an analysis of Wikipedia edits, noting that logs of these edits are a ‘thin’ data source compared to documentary sources in traditional ethnography, and “often do not appear to hold much evidence about the actions they describe” (2011, p. 5). This appearance is because “the utility of such traces does not stem from some inherent documentary quality, but rather because they are produced and circulated within a highly-standardized sociotechnical infrastructure of documentary practice” (2011, p. 5). Understanding such sources then, requires familiarity with the specifics of how this trace data is produced, such as the variety of practices involved in their production. In many situations, such as Geiger and Ribes’s case of Wikipedia editors, trace data is a primary way in which people know one another. Accordingly, incorporating an analysis of this data alongside other ethnographic approaches is a valuable way to understand how people work together.

In this section, I describe how I have collected logged data for analysis. I begin by pointing out that in one part of my research I examined GitHub issues for a specific project in depth. Since issues are conventionally used to report bugs and request new features, they are a fruitful entry-point for identifying breakdowns that reveal how underlying systems operate (Star 1999). My analysis of GitHub issues is described further in Chapter 7, so I will avoid going into detail about that aspect of my research methods at this moment, except to note that this is an example of trace ethnography.

Instead, here I will describe the processes of collecting and analyzing data from IndieWeb’s chat archives and from GitHub repositories for IndieWeb projects. I begin by discussing how data was collected and cleaned, then explain how topics of discussion were analyzed using Latent Dirichlet Allocation (LDA) (Blei et al. 2003), and structures of interaction were studied using social network analysis. For IndieWeb contributors who were active on both chat and GitHub, it was possible to combine analyses to describe their overall role in IndieWeb’s community, at least insofar as could be inferred from that data. After explaining the processes of creating topic models and social networks from this data, I will describe how data sets were combined to identify patterns that occur across these analyses.

In Chapter 1 I referred to another source of logged data about IndieWeb, the social graph of IndieWeb sites hosted at indiemap.org. When I first learned of this dataset, I assumed it would play a significant role in this dissertation’s analysis. However, as I progressed in this dissertation, I realized that IndieMap was not as well suited to addressing my research questions as the data I have analyzed from GitHub and IndieWeb’s chat archives. Specifically, it can provide an overview of how people use their IndieWeb sites but does not offer a vantage to examine how IndieWeb is built. It is true that IndieWeb’s contributors often use their websites as part of their building processes—for example by testing new features on their personal sites, and publishing accounts of their making processes—however, when compared to chat and GitHub data, traces of such activities are less standardized in IndieMap’s data. All of the websites captured by IndieMap use some kind of IndieWeb features, but many have nothing to do with actually building the IndieWeb. Further, because each site can be unique, which is part of the point of IndieWeb, the data does not lend itself to the sort of analysis possible in more standardized chat and GitHub data. I make this comment to explain why I have not analyzed IndieMap’s data to a greater extent, given that

on the surface it may appear to warrant such study.

3.3.1 Collecting and Cleaning Log Data.

Chat log data

IndieWeb’s chat archive is published at chat.indieweb.org, and available for download from a GitHub repository at <https://github.com/indieweb/indieweb-chat-archive>. This archive was downloaded on July 24, 2019. As a result, this analysis includes every archived chat message from Feb 12, 2011 through the time of download on July 24, 2019.

The chat archives are stored as .txt files containing JSON⁶ representations of the chat logs. Using a Python script, the chat archive was converted to a large dataframe with the following columns for each message:⁷

- **datetime:** The ‘datetime’ property, converted from Epoch time (seconds since Jan 1 1970) to a string. Saved as UTC.
- **author_name:** A unique ID to refer to each author by username. This was later cleaned as will be described in the following paragraphs.
- **author_link:** If available, a URL pointing to the author’s personal website.
- **type:** The type of event. Could either be ‘join’ indicating that an individual joined the IRC channel at this moment, or ‘message’ indicating each chat message sent. Only messages were kept for analysis.
- **content:** The content of the message.
- **channel:** The channel to which the message was posted, e.g. #indieweb, #indieweb-dev, #indieweb-meta.

The archived logs are essentially a complete record of discussions in IndieWeb’s chat channels. They are very lightly moderated to remove some spam (especially offensive content) and individual users have been allowed to edit how their messages are logged in two ways:

1. **Typo correction.** Immediately after posting a message, individuals can post a special command to correct a typo. For example, if one accidentally types “*good mrning*” to the chat, they can ensure a corrected version is saved to the logs by posting “*S/mrning/morning*”. The archives will then contain the corrected message, “*good morning*”.
2. **Removing content.** In rare cases, people can request that some information be removed from the logs. In the course of my participant observation, I witnessed this once, when a person accidentally pasted some sensitive account information into the chat. That person asked for the information to be scrubbed from the logs, and one of the chat administrators did so.

Although such minor revisions are possible, I did not observe any substantial edits and thus take the archives to be an accurate representation of IndieWeb’s chat.

⁶JavaScript Object Notation (JSON) is a data format for storing structured data as human-readable text.

⁷Additional columns were collected, but were not used in analysis so are omitted here for clarity.

Table 3.3: Types of events in IRC archives

Type	Description	N
Join	Indicates that a person has joined a chat channel.	655,206
Leave	Indicates that a person has left a chatroom (this is stored inconsistently, and is missing the majority of leave events)	1,964
Topic	Indicates that a channel topic has been changed.	74
Message	Messages posted to chat.	1,159,559

Prior to beginning a formal analysis of IndieWeb’s chat archives, it was necessary to clean them as follows. First, the chat archives contain four types of events, *join*, *leave*, *topic*, and *message*, described in Table 3.3. *Join* and *leave* events seem like they could be useful for identifying the number of people logged into a chat channel at any given moment, but *leave* events were sent inconsistently, making such a count infeasible. Ultimately, only *message* events were of interest, so *join*, *leave*, and *topic* events were removed.

People posting to IndieWeb’s chat sometimes prepend or append their usernames with contextual information. For example, a name might be appended with ‘[laptop]’, ‘[mobile]’, ‘[home]’, ‘[work]’ or other suffixes to indicate which device one is using or where they are posting from. Additionally, IndieWeb’s IRC channels are bridged to a corresponding Slack channel, and *author_names* of people posting via Slack are enclosed in square brackets (e.g. [jackjamieson]). As a result, the same person might be represented by multiple names in the chat archives. Common suffixes were stripped, which merged multiple usernames belonging to the same individual into one unique identifier. Similarly, the format of author personal website URLs was standardized to remove protocol information (*http://* or *https://*) and *www* prefixes.

Known bot accounts were removed from the logs prior to analysis. The most active bot account is a tool called *Loqi*, which performs several functions:

1. Respond to queries by retrieving data from IndieWeb’s wiki. e.g. if a person types “what is *xyz*” in any of IndieWeb’s chat channels, Loqi will look up ‘*xyz*’ on IndieWeb’s wiki and respond with the first sentence from that wiki article.
2. When IndieWeb’s wiki is edited, posts a message describing that edit to the *indieweb-meta* chat channel.
3. Logs Twitter mentions of #indieweb or #indiewebcamp to chat.
4. Saves IRC logs into the archives used for this analysis.

Additionally, some other bots are used to automatically post when specific IndieWeb projects are updated on GitHub, or to perform some other automated processes. Lastly, it was possible to identify and remove messages by some bots that posted by spam. Attempts to identify and remove spam were not exhaustive, and instead focused on identifying frequently repeating advertising messages.

After this cleaning, the number of messages was reduced from 1,159,559 to 923,634. These steps prepared the chat logs for further analysis, which will be described soon.

GitHub log data

A large portion of IndieWeb software projects are hosted on GitHub, a platform for software code and other version-controlled repositories. GitHub repositories include a detailed history of revisions to source code and other documents, logged discussions about bugs, change requests and other topics, and a variety of project management features. By default, GitHub repositories are public, and their data may be accessed either via a Web browser or through GitHub’s API. For this analysis, data was collected through GitHub’s API using the Python package *PyGitHub* (PyGitHub 2020).

IndieWeb is highly modular and has no clear boundaries. Studying collaboration across a multiple GitHub projects that interoperate to contribute to a larger endeavour is one of the contributions of this dissertation. First, it was necessary to identify IndieWeb-related repositories. There is no authoritative list of IndieWeb related GitHub repos. However, IndieWeb’s wiki contains thorough accounts of a large number of projects. Many IndieWeb projects have their own wiki page (e.g. <https://indieweb.org/bridgy>), and others that do not have their own page are often linked from the wiki as examples of IndieWeb practices. Considering IndieWeb’s wiki to be the most complete source for identifying repos, I used this as a basis for compiling a list of IndieWeb-related GitHub repositories. I downloaded a copy of IndieWeb’s Wiki from <https://github.com/indieweb/wiki> on April 17, 2019 at 11:05 AM. Drawing from this archive, a list of IndieWeb-related repos was identified using the procedure described in Appendix D. 1,079 repositories were identified for analysis. The process of identifying IndieWeb-related GitHub repos was not perfect, and there are surely errors of omission, and well as repositories that were identified as IndieWeb-related but who only incidentally refer to IndieWeb keywords. Nonetheless, these provide a reasonable list of IndieWeb related repositories for analysis.

The next step was to collect data from these repositories. Again, using *PyGitHub* to access GitHub’s API, I wrote a Python script to collect event logs from each repo through GitHub’s API. Two types of events were collected: (1) Commits, which record changes to a project’s source code or other files; (2) Events related to issues, which record discussions about feature requests, bugs, and other topics. Events include opening a new issue, commenting, closing an issue, cross-referencing related issues and several other actions. This data was downloaded on May 17, 2019.

After collecting this data from GitHub, I deduplicated redundant events. Duplicates occurred due to a common practice in open source communities of “forking” a project. Forking entails copying a project (including its edit history) and then making changes to that fork. When proposing changes to someone else’s project, it is common to make a fork, implement changes on the new copy of the repository, and then request that your changes be merged back into the original, “master” repository. On GitHub, merging is typically performed using a “pull request,” a mechanism to request that a repository’s maintainer *pull* your changes into the master repository. Forked repositories duplicate commits from their master repository, but do not copy issues. Since the duplicate commits are redundant information, they were identified using their SHA, an encrypted hash identified that is unique for each commit. When duplicate commits were identified, the version from the origin repo was preserved, and copies from forks were dropped.

After removing duplicate commits, the event log contained 104,967 commits and 151,084 issue events (256,051 events total).

3.3.2 Analysis: Topic Modelling

I used Latent Dirichlet Allocation (LDA) (Blei et al. 2003) to identify topics of discussion across IndieWeb’s chat archives. LDA is an unsupervised machine learning technique for identifying latent topics of co-occurring attributes (in the case of text documents, words) in a corpus of documents. Each document is evaluated as a “bag of words,” meaning that word frequency is captured as part of the analysis but word order is not. Topics, as modelled by LDA, consist of collections of frequently co-occurring terms.

The term “topic” is used to represent an intuitive comprehension of these collections of terms. However, there is no inherent epistemological claim in LDA that topics represent anything beyond probability distributions of latent multinomial variables in sets of words. Therefore, to make a claim that these variables represent topics in a meaningful sense, they must be subject to evaluation.

By identifying terms that frequently co-occur in the same document, LDA is useful for identifying unexpected topics. Additionally, it makes it possible to assign topics to documents even if they do not contain a pre-determined keyword. In the case of IndieWeb’s IRC logs, this is especially useful for identifying texts related to IndieWeb’s principles, since explicit ‘principle’ keywords (i.e. key terms from indieweb.org/principles) occur relatively rarely even in relevant conversations.

LDA was conducted on these documents using Mallet (McCallum 2002), a Java package for a variety of machine learning applications related to text. Mallet was run within a Python Jupyter notebook via Gensim (Rehurek and Sojka 2010), a package for implementing various topic modelling methods. Gensim can perform LDA using its internal scripts, but early tests demonstrated that Mallet consistently created models with a higher coherence than Gensim’s data.

Pooling data.

One challenge posed by applying LDA to IndieWeb’s IRC logs is that individual messages are short and meaning may be conveyed across multiple sequential messages. Short and messy texts have been observed to lack sufficient information to assign a topic (Nguyen et al. n.d.). One method of addressing this challenge is to train a topic model from an external corpus. For example, Phan et al. (2011) used Wikipedia as a large corpus to train a topic model for analyzing sparse documents. Another approach was proposed by Mehrotra et al. (2013), who pooled short texts into larger documents and found this improved the coherence of topics identified by their models.

Because IndieWeb’s chat includes a lot of IndieWeb-specific terminology and other technical jargon, most external corpora would be too general to be useful in this case. One possibility would be to use the text from IndieWeb’s wiki, which includes similar language as its chat but is more structured and consists of longer documents. However, I chose to pool multiple chat messages together, in part because this removed the necessity of collecting and processing IndieWeb’s wiki articles, and because even though both chat and the Wiki contain similar technical terminology, they differ significantly in tone and style.

I attempted multiple types of pooling chat messages to build a suitable topic model. Although individual messages are often short and messy, conversations are typically held over multiple messages in a short time period. As a result, messages that are sent by a single user in a short time frame are likely to be related as part of the same larger conversation. To keep related messages together, individual chat messages were grouped using the following criteria:

1. **No grouping.** Each message is a separate document.

2. **Group by time and channel.** Each document pools all messages posted to a channel in a 30-minute time period. This was attempted to see if it could better account for multiple people talking about the same topic together.
3. **Group by time, author, and channel.** Each document pools a single person’s messages to one channel in a 30-minute time period.

These were evaluated using the coherence measure built into Gensim’s bridge to Mallet for LDA, which is an implementation of the four-stage topic coherence pipeline from Röder, Both, and Hinneburg (2015). Results of this evaluation are presented in Figure 3.1. The third pooling method, in which messages were pooled in 30-minute chunks by both author name and chat channel, consistently demonstrated higher coherence by this metric, and was thus selected for this analysis.

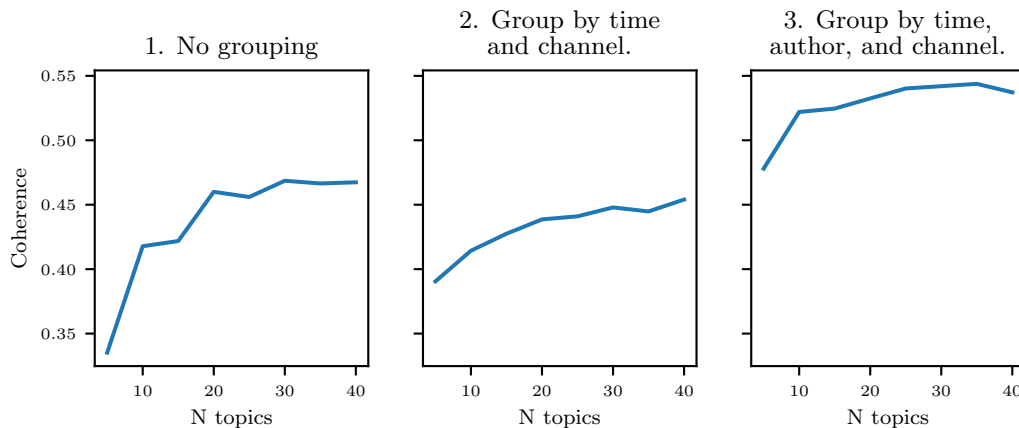


Figure 3.1: Comparison of coherence scores for different pooling methods.

Once grouped, the initial 923,634 messages that were present in IndieWeb’s chat archives were reduced to 210,291 documents.

Text cleaning.

Prior to LDA, documents were cleaned by converting all text to lower case, then removing special characters, punctuation, and stop words. spaCy (spaCy 2020), a natural language processing library for Python, was then used to lemmatize words, finding the base form for each word so multiple tenses and variations of each word were grouped.

Finally, before generating models, I randomized the order of documents to prevent topics established early in the archives from having an outsized influence compared to those that emerged later.⁸

Selecting number of topics.

When performing LDA, it is necessary to specify the number of “topics” before training the model. In most cases, it is only possible to evaluate the most suitable topic count after the model has been trained. Therefore, multiple models were trained, with topic counts ranging from 2 to 30. The first stage of evaluation was to calculate the coherence score for each model, visualized in Figure 3.2.

⁸This was based on a suggestion by Brenda Moon, who shared her insights about LDA at a conversation during the *10th International on Social Media & Society* in 2019.

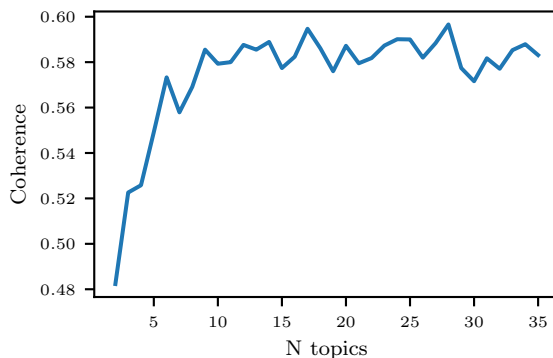


Figure 3.2: Topic model coherence by topic count.

Quantitative coherence scores are useful for assessing overall fit of a model and selecting an optimum number of topics to assign to a model (Prabhakaran 2018). For example, if too few topics are selected, the model is likely to return poorly coherent topics that combine multiple distinct concepts and terms. As one increases the number of topics, coherence will increase and eventually reach a plateau. Topic counts below that plateau are likely to be poorly fit compared to topic counts above that plateau. After reaching that plateau, topics will become increasingly precise, yet may become difficult for humans to interpret if they become too fine-grained (Chang et al. n.d.; Moon 2016). Given this quality, beyond a minimal level of coherence, the number of topics must be selected based on how effectively the topic model achieves its purpose. In this study, my purpose is to generally identify subjects of conversation as a means to evaluate the extent to which individuals are active in conversations about IndieWeb’s principles as well as other broad-stroke topics such as events and event-planning, online community maintenance, and IndieWeb-specific technical objects. Accordingly, my goal is to select the smallest number of topics that can achieve this goal.

Figure 3.2 shows that coherence begins to level off at around 6 topics and is relatively flat from 10 topics onward. To select a topic count, models with 6, 10, 16, and 30 topics were analyzed further, as described below:

1. I manually generated lists of keywords associated with specific topics that were of interest, and that I expected to emerge as latent topics through LDA. These keywords concerned IndieWeb’s principles (drawing from indieweb.org/principles), IndieWeb building blocks such as standards and apps, and terms related to events. A full list of these terms is presented in Appendix E. I then evaluated which models most effectively grouped these keywords into the same topic, with an expectation that, for example, event or principle-related keywords should generally be grouped together.
2. For the most viable models identified in the previous step, I reviewed the most representative documents for each topic and assessed its meaning qualitatively.

These two steps provided a means to identify LDA models that were consistent with my intuitive understanding of the topics, and to build confidence in these models.

The model with 16 topics was selected. Having selected this model, each document was assigned a dominant topic, which is the topic that most strongly contributes to that documents meaning. Each topic

was then labelled based on qualitative assessment of its most common keywords as well as re-reviewing both its most representative documents and a random selection. `pyLDAvis` (Mabey 2018), a python port of the R package `LDAvis` (Sievert and Shirley 2014), was used to visualize the distribution of keywords in each topic. `pyLDAvis` includes an option to adjust how the relevance of terms to their parent topic is calculated by modified the λ value used in this determination. Sievert and Shirley (2014) note that human accuracy in categorizing topics was best when λ was 0.6, and I followed this recommendation. Upon labelling the topics, it became evident that some overlapped and could be grouped together.

The resulting classification of topics is presented in Chapter 5 Section 5.2. Additionally, the results of topic analysis were integrated with a social network analysis, the methods for which are described in Section 3.3.3.

3.3.3 Analysis: Network Analysis

Online communities are often well suited to social network analysis, since interactions such as messages are recorded as discrete units which are often explicitly related as social networks. I use social network analysis to identify large-scale patterns of influence in IndieWeb’s chat logs and in interactions on GitHub. I describe the processes of constructing networks from these datasets, and then identify metrics used in their analysis.

Building a network from chat archives

Unlike more structured social networking logs, such as Twitter data, which is explicitly linked as sequences of replies, retweets, favourites etc., messages in IndieWeb’s chat logs do not contain structured information about their intended recipients. Nonetheless, as described in previous studies of IRC (Mutton 2004; Tuulos and Tirri 2004), recipients of each message can be inferred using heuristics based on explicit references to an individual’s chat nickname or the time between messages.

Using a Python script, I identified messages that included reference to another individuals’ chat nickname. I then defined an edge — a connection among two nodes — with the following information:

- **Message ID:** A unique identifier for the message containing a direct mention.
- **Source:** The person who sent the message.
- **Target:** The person mentioned in the message.
- **Mention Edge:** A Boolean value indicating that this was a direct mention.

Some messages mention multiple people, in which case multiple edges could be created for that message. There was potential for false positives, especially if a target’s username was a dictionary word. To reduce such errors, usernames that were dictionary words were excluded if they had fewer than five total posts to the chat.

The second type of edge was based on temporal proximity. When two people post to the same IRC channel around the same time, there is a reasonable likelihood they are conversing with one another, or if they are merely having two separate but parallel conversations are likely to be reading each other’s messages. Thus, if two messages are posted in quick succession, the second is considered a potential reply to the first, and an edge is created to represent this.

The maximum time between messages for them to be considered part of a conversation was defined based on Kalman et al.'s (2006) work on chronemic cues during asynchronous written computer-mediated communications. They examined response latency between messages and found that a latency of greater than 10 times the average was likely to indicate a lack of reply. Their study examined communications via Instant Messaging (IM) software. The main difference between IM communications and the chat logs examined in this study is that IM is typically used for one-to-one discussions, whereas chat is used for discussions among groups. Nonetheless, this metric provides a reasonable approach for identifying messages as part of conversations.

The time between each message was calculated, which was then used to calculate a median. The median interval between messages varied significantly between channels and over time. This likely reflects differences over time such as (a) how many people were participating in each channel, (b) what proportion of participants were regulars compared to new or infrequent posters and (c) shifting patterns of desktop vs mobile use from 2011 through 2019. Across each combination of *channel* and *year*, the briefest median interval was 17 seconds, the longest was 105 seconds, and the median median interval was 31 seconds. To account for this variance, the maximum interval for considering two messages to be part of the same conversation was set at 10 times the median for each channel/year combination.

When messages were linked based on temporal proximity, I defined edges with the following information:

- **Message ID:** A unique identifier for the message containing a direct mention.
- **Source:** The person who sent the message. (User 2)
- **Target:** The person to whom the message was directed. (User 1)
- **Proximity Edge:** A Boolean value indicating that this edge was defined according to temporal proximity.

This method has potential for both false positives and false negatives. False positives can occur if messages occur in close proximity but are not part of the same conversation. Messages from such parallel conversations are likely to be at least read by everyone posting at the same time, and therefore I do not consider this to be a major problem. False negatives can occur when people take a long time to respond, such as if someone is busy doing something else or away from their keyboard. Sometimes, but not always, when replying after an extended pause, it is common practice to direct mention the person they are addressing, so the use of direct mention edges as described above mitigates this problem.

Finally, there are instances where a single message can form two edges between the same two nodes, both temporal proximity and by direct mention. In such cases, the two edges are collapsed into a single edge.

Because each edge is associated with a specific message, edges can be associated with the message's timestamp and content. Moreover, since the previously described LDA model labels each message with a topic, it is possible to combine network analysis with the topic model discussed previously.

Building a network from GitHub events

A social network was also built from the GitHub event data, incorporating methods and approaches from recent studies of open source developer networks on GitHub (Hu et al. 2018; Strzalkowski et al.

2019; Cheng and Guo 2019, e.g.). Based on Cheng and Guo’s (2019) analysis of roles and dynamics in GitHub contributions, I grouped some actions together into *commits* — changes to code or files, *issues* - opening or commenting upon issues, *networking* - linking related issues and/or commits to one another, and *admin activities* - marking events as complete or assigning labels to events. Additional events from the GitHub logs, such as subscribing to a repository, adding project milestones, and others were excluded from the analysis.

This was a simplified version of the taxonomy used in their study. A notable simplification is that, unlike Cheng and Guo, I did not evaluate commits or other contributions by length. I worry that length of messages can appear to offer more precision about degrees of contribution than is valid. For example, a very long issue comment might include pasted output of an error log, and a much shorter comment might be a qualitatively more substantial opinion contribution. Similarly, commit length is a rough metric of code contribution, but shorter commits can be more substantial than long commits, especially when comparing across repos that are written with varying degrees of verbosity. Second, I do not include pull requests as a separate category of action. Pull requests are an important part of open source collaborations on GitHub, but for my purposes, their data is duplicated by issues and commits. (In GitHub’s API, pull requests are simply issues with commits attached, therefore they are already covered by data about commits and issues).

Unlike the chat data, in which edges needed to be inferred, relationships between events and objects are explicitly identified in data collected from GitHub. Specifically, issues, commits, and related actions are performed by a user, and targeted toward a specific repository. Having identified nodes and edges, a network was created using *NetworkX* (NetworkX 2019), a python library for network analysis. The collected event logs were represented as a bipartite network graph. Each node represented either a user or a repo and edges represented interactions from a user to a repo.

3.3.4 Linking chat and GitHub data

A perennial challenge for researchers working with communication data is accounting for *media multiplexity* (Haythornthwaite 2001). People routinely communicate across multiple channels, such as telephone, email, and in-person. This challenge is intensified in the age of platforms, which may offer specialized communication tools for specific purposes. This is evident in IndieWeb, where communication occurs on IRC, IndieWeb’s wiki, in-person at events, through GitHub, through personal websites, and many other channels such as personal conversations through Slack, Google Hangouts, etc. The network and topic analyses described here are not feasible across all of those communication channels. However, combining data about both chat and GitHub contributions provides a means to study a significant body of IndieWeb’s online activity.

Common identifiers.

Network data about GitHub contributions was linked with network data and an LDA model about IndieWeb’s chat archives. Many individuals who contribute to IndieWeb participate at both sites, and identify themselves at both using the same personal website URL. Therefore, website URLs were used to link individuals at both sites, even if they use a different username at each. For example, my chat username is ‘jackjamieson’ and my GitHub username is ‘jackjamieson2,’ but both profiles are associated with my personal website URL `jackjamieson.net`.

IndieWeb’s wiki includes a page called *indieweb.org/chat-names*, which is used to link together IRC usernames and profile pages on IndieWeb’s wiki. Individuals can add themselves to the */chat-names* page if they want to include information from their wiki profile in IndieWeb’s chat log archives at *chat.indieweb.org*. Although not every participant completes this section, active IndieWeb IRC contributors are likely to. Since this list does not include every person who posts to IndieWeb’s IRC channels, but rather those who take the step to register on the *indieweb.org/chat-names* page on IndieWeb’s wiki, this is likely to identify core users. In GitHub, personal website URLs are indicated on as part of each user’s profile. A python script was used to fetch the *blog_URL* property from each person’s profile through GitHub’s API.

There are many cases where an individual from IndieWeb’s chat could not be identified on GitHub and vice versa. This could result when an individual did not participate at one of the sites during a given time period, or if the process used to cross-reference across these sites failed. Since personal domain names are considered the centre of one’s online identity on the IndieWeb, this process was effective at identifying a large number of links. However, it errs towards individuals who are relatively dedicated to IndieWeb ways of doing things and likely excludes many peripheral contributors.

To avoid including a large amount of GitHub activity that was unrelated to IndieWeb, only observations that included IndieWeb chat activity were included in this analysis. This means some individuals are included who *only* participated in IndieWeb’s chat, and who did not use GitHub (or whose use was not observed).

Common units of time.

Interactional data from IndieWeb’s chat channels and from GitHub are timestamped. To be able to understand relationships among these two data sources, it necessary to determine a temporal framing that is commensurable across both. Accordingly, I have grouped events at both sites into monthly periods. This makes it possible to interpret and compare patterns that unfold across time, such as growth and contraction of the network, individuals’ longevity of participation, shifting topics of conversation, and relationships between activities on IRC and contemporaneous activities on GitHub.

To make my analysis of GitHub compatible with the social network analysis of IndieWeb’s chat logs, the bipartite network was projected into a graph of users, connected by shared participation to the same repo within the same monthly period. Organizing these logs into monthly chunks created a basis to evaluate user-level and network-level metrics from each dataset for each month, and then combine these into one large dataset.

Network-level analysis

A network-level analysis was conducted in which each observation summarized one month of time across the entire network. The following metrics were calculated:

- **Number of chat messages:** The total number of messages posted to chat.
- **Number of chat users:** The number of unique users who posted at least one message to chat.
- **Chat topic shares:** The portion of chat documents assigned to each topic group (each document consisting of pooled chat messages as described in Section 3.3.2). This is intended to evaluate patterns where the entire network’s topics shift in prevalence over time.

- **Number of GitHub events by type:** The total number of commits, issues, network actions, and admin actions made to IndieWeb-related GitHub repos.
- **Number of GitHub users:** The number of unique users who interacted with IndieWeb-related GitHub repos.

These network-level metrics provide a general overview of how and what people were discussing in IndieWeb’s chat, as well as how they worked together on GitHub.

User-level analysis

Additionally, user level metrics were calculated for each month. Each observation under this analysis represents one users’ activity over a time period of one month. The following metrics were used:

- **Number of chat messages:** The number of messages each person posted to chat.
- **Chat betweenness centrality:** Each person’s betweenness centrality. Betweenness centrality measures the extent to which each node falls on the shortest path between other nodes (Freeman 1977). Nodes with high betweenness centrality are likely to be influential, since they are conduits through which information can be shared with otherwise unconnected nodes. The chat data describes a temporal network in which edges among nodes are created in chronological sequences, and I account for temporality when defining betweenness centrality. As Lambiotte et al. (2018) explain, “For a sequence of two links $A \rightarrow B$ and $B \rightarrow C$, a node A can only influence C via a (transitive) *time-respecting path* via B if $A \rightarrow B$ occurs before $B \rightarrow C$. Such a transitive path from A to C does not exist if the ordering of links is reversed” (emphasis in original). To address this, betweenness centrality was calculated using *pathpy*, a Python package for doing higher-order network analytics, which accounts for temporal sequences the purpose of node centralities and other network measures (Data Analytics Group, University of Zurich 2019).
- **Chat topic shares:** The proportion of each observations’ summed topic probability distribution allocated to each topic. This was calculated by summing each topic’s probability scores, then dividing by the sum of all calculated probability distributions:

$$\text{Topic share} = \frac{\sum \text{This topic's probability scores}}{\sum \text{All topics' probability scores}}$$

- **Number of GitHub events by type:** The total number of commits, issue events, network events, and admin events made to IndieWeb-related GitHub repos.
- **GitHub betweenness centrality:** Unlike the chat data, where *pathpy* was used to account for temporality when calculating betweenness centrality, the nature of the GitHub data made it necessary to evaluate only an overall centrality for each month. This is because precise timestamp data was lost in the process of converting the bipartite user-to-repo network into a projected user-to-user network. Therefore, GitHub betweenness centrality was calculated using NetworkX.

These user-level measures provide a basis for describing participants’ roles during different months.

Next, an agglomerative (bottom-up) hierarchical cluster analysis was used to identify roles of IndieWeb contributors each month based on the topics they discussed in chat and the types of activities

they engaged in on GitHub. Hierarchical clustering does not require a pre-determined number of clusters, unlike flat methods such as K-means. This was appropriate for the exploratory nature of this analysis, in which I sought to identify if multiple variables could converge, defining ‘roles’ based on combinations of topics used together, proportions of different GitHub activities, and other variables.

As described in Section 3.3.2, some topics were grouped together. This contributed to an imbalance in topic scores where groups consisting of multiple individual topics had higher values than groups consisting of only one topic. This led to an accurate representation of the data in that topic groups consisting of multiple individual topics did account for a large portion of discussion. But, for the purpose of clustering, this resulted in almost every observation being categorized with a single dominant topic. A more useful model would emphasize variance among individuals’ behaviours, which requires balancing the variables used for clustering.

To address this challenge, topic scores were rescaled in two steps:

1. **Mean-centering.** Topic scores were mean centred using the following formula:

$$\text{mean centred value} = \frac{\text{value} - \text{topic mean value}}{\text{topic standard deviation}}$$

This standardizes all of the topic scores to the same scale.⁹

2. **Normalization.** Topic scores are then processed using the `normalize` function from `scikit-learn` (`scikit-learn` 2020). This scales the mean-centered topic scores for each observation (representing one month of one individual’s data) to have unit norm. As a result, each observation is transformed into a proportion of the whole, to indicate that, for example, 50 per cent of conversations were about topic 1, 25 per cent about topic 2, and so on.

Once topic scores were re-scaled, I clustered the data in two ways. First, I created clusters defined by topic shares. This resulted in a variable identify each user’s dominant topic in each month. Second, I created a cluster that classified each users’ activity on GitHub over each month. The culmination of this user-level analysis is a set of variables for summarizing the activities performed by each individual in a given month, which allows me to identify relationships between chat and GitHub activity.

3.4 Combining qualitative and computational analyses

This chapter has described multiple methods that I used for studying IndieWeb. The results are discussed through the next four chapters. By combining multiple methods, my intention is to investigate the processes involved in building a system like IndieWeb’s, while attending to multiple scales through which influence and action operate. I draw on interviews, observation, and reflections on making my own IndieWeb to describe the experience of building for the IndieWeb in Chapter 4. The following two chapters focus situate that experience in IndieWeb’s community. Chapter 5 uses large-scale analyses of logged interactional data about IndieWeb’s chat and GitHub activities to describe a high-level overview of the community structure. This analysis illustrates the scale of IndieWeb’s community of builders and identifies a centre of influence, but cannot thoroughly explain who is included or excluded from this centre or why. To address that limitation, Chapter 6 presents interview participants’ experiences and

⁹Thank you to Chang Lin at the Toronto Region Statistics Canada Research Data Centre for advice about scaling this data using mean-centring.

perspectives of influence and exclusion in IndieWeb’s community, as well as efforts to address potential and observed barriers. Finally, Chapter 7 uses trace ethnography (Geiger and Ribes 2011) and interviews to investigate how IndieWeb’s syndication relationship with the “corporate web” influences development and maintenance.

In trying to work across epistemic boundaries this dissertation encounters its own epistemic obstacles in terms of methods. Computational analyses present the world as universal in a way that clashes with the assumptions and findings of smaller-scale qualitative inquiries. These challenges form a set of productive tensions that must be considered while presenting and discussing the results of these analyses, and which is discussed further in Chapter 8. Actually engaging with these tensions can be an important step toward bridging the “great divide” between academic disciplines (G. Bowker, Star, Turner, et al. 1997).

3.5 Ethical considerations

Having described this dissertation’s processes of data collection and analysis, I will now review ethical considerations. I place these toward the end of this chapter not because they are an afterthought, but instead so these matters can be discussed in context with the multiple data used in this project.

First, using interactional log data requires particular ethical considerations. Methods such as interviews are preceded by affirmations of informed consent, and participant-observation includes opportunities (or depending on the context, requirements) for researchers to disclose the nature of their data collection and analysis. By contrast, analyses of public logs of interactional data present obstacles for obtaining informed consent and disclosing researchers’ activities to all participants. In this case, data collected from IndieWeb’s chat channels and IndieWeb-related GitHub repositories involves thousands of participants, many of whom are no longer active and are not reachable for consent purposes.

This circumstance is common in studies of social media, where researchers have routinely collected large quantities of tweets and other public posts for analysis. Questions of ethics about using such data are not easily settled. One school of thought views information publicly shared on social media platforms as suitable for researchers without needing informed consent (ESOMAR 2011, e.g.). Wilkinson and Thelwall (2011) note that the default position in studies of Internet documents has often been that text authors should neither be asked for consent nor informed that their text is under study, a drastic difference from traditional social science research. However, they note that even in studies that rely on publicly accessible online data, efforts are made to keep text authors anonymous (2011).

Another school of thought views the use of online data more critically. boyd emphasizes that “Just because it is accessible doesn’t mean using it is ethical” (boyd 2010). When someone posts data to a publicly accessible platform or site, they are not necessarily consenting to its aggregation or analysis in a different context. In fact, when multiple people participate in a “public” online conversation, they may each have different understandings and expectations about the appropriateness of that data being used by researchers. A 2014 study, consisting of focus groups and interviews with 34 people, found a range of views about the appropriateness of using social media data for research (Beninger 2017). With regards to informed consent, some participants felt that online content was inherently public, while others felt that researchers *should* seek consent but acknowledged that this would be logistically impractical

In a presentation at the *2019 International Conference on Social Media & Society*, Burkell and Regan (2019) argued that common research methods for using online data are not ethically defensible. They assert that these methods tend to extend from traditional observational research where ethical

standards rely on three criteria: (1) The behaviour being observed is public, (2) Research subjects have no reasonable expectation of privacy, and (3) individuals are not made identifiable by the research. Online observational data may initially appear to meet these standards, but it is not always straightforward to address these criteria upon further examination. As a result, public/private is not a straightforward distinction, but should be considered on a continuum. Based on this challenge, Burkell and Regan (2019) identify four criteria that make online data *more* public:

1. Research use is allowed.
2. The group is open.
3. Discussions are archived
4. Discussions are indexed on the open Web.

These criteria align with considerations I made before beginning this study, which I will elaborate upon briefly.

Research use is allowed. Prior to my participation, other researchers have been welcomed to study the IndieWeb. Notably, Amy Guy’s (2017) dissertation discusses IndieWeb (alongside other Web systems) and her work included archival analysis of IndieWeb’s wiki. During her research, she was an active and welcome member of IndieWeb’s community, and her status as a researcher was well understood. Another scholar, Greg McVerry, is currently engaged in a study of the IndieWeb, and is similarly welcomed as an active member of the community. And most importantly, I have disclosed my status as a researcher at IndieWeb events and online, and similarly been welcomed with no reservations expressed.

The group is open. The sources from which I have drawn documentary data are fully open, with no login required to access them. Similarly, the events and online sites through which I have engaged in participant-observation are open to the public.

Discussion are archived. The sources from which I have collected logged interactional data are publicly archived and participants are made aware of this fact. For example, upon joining one of IndieWeb’s chat channels, a message is displayed with information about the purpose of the channel. With the exception of the *indieweb-chat* channel, which is not logged, this topic message includes a link to the published chat logs. For example, the main *#indieweb* channel includes the message: ‘log <https://chat.indieweb.org/today>.’ Given this message, it is reasonable to conclude that participants are aware that their conversations are archived. Since the *indieweb-chat* channel is not archived, it has not been included in my analysis. Similarly, GitHub repositories are archives; The general purpose of a public GitHub repository is to preserve a record of discussions and changes to code and make it accessible to anyone who may be interested. Similarly, IndieWeb’s wiki is obviously a public archive, and its contents are released to the public domain under a creative commons license.¹⁰

Discussions are indexed on the open Web. All data that I have considered to be “public” for the purpose of this study is accessible on the open Web. No permission or login is required to find or access this data. In most cases, the data is directly indexable by search engines. In some cases, such as meeting minutes logged using Etherpad¹¹, I believe the documents are not directly indexed by search engines but are linked from pages on IndieWeb’s wiki which are indexable.

¹⁰See copyright notice for IndieWeb’s wiki at <https://indieweb.org/IndieWeb:Copyrights>

¹¹Etherpad is a Web-based collaborative text editor that is used to record session notes during IndieWeb events and similar public discussions

Additionally, people who participate in IndieWeb-related discussions tend to have a strong understanding of the publicity of online data. This is particularly so for individuals who participate in the construction of IndieWeb tools, debates about IndieWeb’s values, and related topics. Given the nature of IndieWeb, the people involved in this study generally have a strong awareness of how data is stored and used on the Internet, so there is little chance they are unaware that their data is public, archived, and indexed on the open Web.

Nonetheless, I endeavour to protect participants’ anonymity where appropriate and possible. Even with IndieWeb’s chat logs being publicly archived, I am cautious about presenting direct quotations. First, where direct quotations are used, it is not feasible to anonymize the quotation because the chat archives are searchable. And even though contributors to IndieWeb’s chat, wiki, GitHub repositories and other resources have implicitly accepted that their data may be accessed by anyone, this does not mean they automatically consent to being quoted in research publications. Accordingly, although I assert there is no ethical requirement for informed consent prior to using this material as a means of understanding the IndieWeb community, in the case of direct quotations from IndieWeb chat and GitHub, I have asked for the authors’ permission in advance of publishing this dissertation.

This is a particular consideration for communications in chat rooms and GitHub issues. Even though these conversations are logged publicly, they are often uttered spur of the moment and their interpretation may require surrounding context. In fact, one of the participants whom I asked about using quotations from chat voiced a concern about context collapse. Many of the conversations in IndieWeb’s chat are part of a larger conversation that spans across time and multiple media (chat, wiki, in-person conversation, blog posts, etc.). To address this concern, I have offered to send direct quotations to participants in advance so they can validate my presentation of their statements. This does not entail softening my *interpretations* of broader patterns and trends (unless I learn of a significant blind spot) but instead aims to ensure I do not inadvertently harm participants’ reputations by amplifying public statements without necessary context.

When quoting from participants websites, I have exercised judgement about the potential for context collapse. Some people use their websites as public figures, publishing polished editorials and articles that are clearly intended to stand on their own. I have not sought permission to quote from website articles that are clearly intended for public consumption. However, in every instance where a quotation or other statement in this dissertation identifies an individual, I have carefully considered potential harms and used my best judgement.

Finally, in every interaction related to this project, I have abided by IndieWeb’s code of conduct, which applies to “all IndieWeb spaces both online and off, including our events, IRC channel, and wiki” (IndieWeb.org 2019d). The code of conduct is summarized as “Be respectful of other people, respectfully ask people to stop if you are bothered, and if you can’t resolve an issue, contact staff. If you are being a problem, it will be apparent and you’ll be asked to leave.” The full text is accessible at <https://indieweb.org/code-of-conduct>.

3.6 Limitations

Since there were no in-person IndieWeb events near Toronto, attending events in-person was expensive and time-consuming and so I only attended IndieWeb Summits. Remote attendance was a good alternative, however scheduling conflicts meant it was only possible to remote attend a limited number of

events. I was able to catch up on other events after the fact by watching videos and reviewing archived session notes, but this is limited compared to attending live. As a result, at times my analysis risked understating the importance of these events compared to day-to-day development as captured in chat discussions and on GitHub. Interviews helped identify and address this limitation, though overall this dissertation focuses on activities that were visible in online archives. This is likely to highlight the contributions of more central community members, who regularly participate online and document their work, and diminish the importance of newcomers at events and others whose primary interaction with the IndieWeb is in person.

This is particularly a risk in analyzing logged interactional data, since both network analysis and topic modelling are affected by quantities of content. For example, measures of network influence such as centralities favour nodes with many connections. This is likely to understate the role of individuals who are quietly influential, such as people who post rarely but with great impact. Interviews help to assuage this to some extent, though interview recruitment, too, had a bias towards those who are already vocal members of IndieWeb's community.

Chapter 4

Building for yourself: Creating IndieWeb tools with values in mind

4.1 Introduction

Chapters 4 through 7 present findings from my analyses. In this chapter, I begin to address **RQ1:** *What values are central to the IndieWeb, and how are they defined, selected, and critiqued?* and **RQ2:** *How are values employed in design activities?* Methodologically, this chapter presents results from my participant-observation and interviews, including a description of my own experiences building IndieWeb software. I focus on individual and small collaborative acts of building, describing design decisions, motivations, and outcomes. By beginning at this scale, I aim to reflect IndieWeb's commitment to individuals building first and foremost for their own needs. In the course of this analysis, I identify ways in which individuals' experiences are impacted by structures, relationships, and design decisions across IndieWeb's community and larger infrastructures. Thus, this chapter establishes a foundation of what building for the IndieWeb is like as an individual, which will then be extended upon in Chapters 5 and 6 (focusing on IndieWeb's community) and Chapter 7 (focusing on IndieWeb's relationships with centralized Web platforms).

Ultimately, individual websites are the central unit of the IndieWeb. Individuals are encouraged to build for themselves, and adoption of their ideas or code on a larger scale is a generally positioned as a secondary goal. First and foremost, people building for the IndieWeb are encouraged to meet their own needs. As a result, IndieWeb is presented as heterogeneous and individualistic by design because each website can be running different software and have different features. Yet, IndieWeb's network of diverse and varied websites demonstrates cohesion as a larger system. Even though each website can be different, the basis of the IndieWeb is that they can communicate with each other and interoperate with IndieWeb applications. This requires mutually defined structures of communication. Thus, in the process of investigating how values are employed in IndieWeb's design it is an important premise that design activities are distributed and independent. Yet it is equally important to understand how individual designs adhere to a shared repertoire of conventions and standards. In this chapter I argue that IndieWeb's technical standards and building blocks support building approaches that balance commensurability and independence.

I begin by investigating overarching values in IndieWeb's goals related to autonomy and self-empowerment.

I then discuss how values are further specified in IndieWeb’s principles for building. In doing so, I address how values are employed in design activities by showing how they are integrated with expected outcomes and recommended processes for building. Following that, I identify technical standards and IndieWeb’s ‘building block’ approach as technical ties through which individuals’ activities are coordinated. This is a co-constitutive process wherein individuals collectively develop a technical infrastructure, which in turn defines conditions in which future IndieWeb development occurs. Finally, I present a description of my experience building an IndieWeb software called *Yarns*. Working on *Yarns* exposed me to the actual work involved in building for the IndieWeb, and particularly made visible a set of tensions between pursuing my individual goals and achieving interoperability with other IndieWeb software, which led to both technical and philosophical compromises.

Through this account, I identify ways in which IndieWeb’s commitments to individual autonomy are sometimes in tension with requirements for consensus and coordination. I argue that both sides of this tension are expressed in IndieWeb’s principles, norms, and standards, the designs of which generally attempt to maximize individual autonomy at first, and fall back to collectivist approaches when deemed necessary.

4.2 Autonomy and self-empowerment as guiding values

Since I am concerned with how values are employed in the construction of a technical infrastructure, I begin with an account of *what* values are declared to be important by IndieWeb’s community. According to its website, “the IndieWeb is a community of individual personal websites, connected by simple standards, based on the principles of owning your domain, using it as your primary identity, to publish on your own site (optionally syndicate elsewhere), and own your data” (IndieWeb.org 2018a). Three reasons are presented for using the IndieWeb instead of corporate platforms: “Your content is yours”; “You are better connected”; and “You are in control” (IndieWeb.org 2020c). A more detailed explanation on IndieWeb’s wiki in an article titled “why” (IndieWeb.org 2020l). This page presents a collection of links and statements describing various people’s reasons for having an IndieWeb site. Examples include:

“On my blog I have control & agency. Full server logs, ability to block abusive referrers, control of comments.”

“The freedom to decide what content and what types of content to publish.”

“Having a blog can get you through the tough times, the lonely times and the confusing times.”

“Maybe you got banned from Facebook for using a ‘fake name’, even though you’d already proven to them that it was your ‘real’ name.”

“You aren’t happy with the community or perceived community that comes baked in with the silo-based tools you use to publish. Perhaps you want to share photos of things you like without people making assumptions regarding your gender or race or social class (see danah boyd’s *The Not-So-Hidden Politics of Class Online*).”¹

(Selected excerpts from IndieWeb.org (2020l))

¹This includes a link to boyd (2009)

In 2014, Tantek Çelik posted a message to IndieWeb’s chat stating that “all of [the ‘why’ page] can be summarized in one word, and that’s autonomy.”² Further, when presenting an opening keynote at IndieWeb Summit 2016, Çelik asserted that IndieWeb’s commitments to owning your own data, being connected, and being in control all stem from one goal: “self-empowerment” (Çelik 2016).

These terms provide sensitizing concepts to guide further empirical investigation. Blumer (1954) explains sensitizing concepts as follows:

A sensitizing concept lacks such specification of attributes or bench marks and consequently it does not enable the user to move directly to the instance and its relevant content. Instead, it gives the user a general sense of reference and guidance in approaching empirical instances. Whereas definitive concepts provide prescriptions of what to see, sensitizing concepts merely suggest directions along which to look. (p. 7)

These overarching goals provide a suitable entry-point to understand the kind of values that motivate IndieWeb contributors. Before proceeding to a more specific description of IndieWeb’s motivations, I discuss how *autonomy* and *self-empowerment* have been described in scholarship about communities similar to IndieWeb.

4.2.1 Empowerment and making

Roedl, S. Bardzell, and J. Bardzell (2015) performed a discourse analysis of 191 papers related to maker culture and discussed how the concept of *empowerment* was employed.³ Before discussing their review, it is worth briefly acknowledging that Roedl, S. Bardzell, and J. Bardzell (2015) refer to “empowerment” and do not use the phrase “self-empowerment.” However, their account mostly identifies rhetoric of empowerment arising from the self, and I am concerned with this facet of the term. Empowerment is framed as a consequence of the skills and abilities embodied in material relationships to technology (2015, p. 9), which are described across three claims. First, makers are empowered because they are able to customize and build upon what most would view as “finished products” to suit their purposes. By “transcending the intended design of a product,” makers are empowered with an elevated degree of control over products (2015, p. 10). Second, extending from this first theme, by customizing technologies to suit their needs, makers can extend the useful life of technologies by repairing them or reusing them in novel ways. This contributes to empowerment in that makers can “avoid having to repeatedly discard devices and purchase new ones” (2015, p. 10). Third, engaging in making can enhance the personal meaning of objects and nurture an “ethic of long-term care” (2015, p. 10). The authors summarize that a recurring theme in literature about making communities is that “making is imagined as personally empowering in the sense that it facilitates a deeply satisfying relationship to objects that aid in a cultivation of one’s identity.” (2015, p. 11)

In addition to these accounts of material-empowerment, Roedl, S. Bardzell, and J. Bardzell (2015) describe how participants in maker culture have been characterized as socially progressive subjects. In the literature they reviewed, engaging in making was frequently described as pleasurable and appealing, and as result was framed as a potential means for technical literacy and redistributed social power: “The common argument is that if HCI acknowledges the DIY movement and begins to design for ‘makers’

²<https://chat.indieweb.org/2014-01-16#t1389841785000000>

³Maker culture typically is associated with communities that build physical objects, such as electronics, 3D printed goods, or traditional arts and crafts. However, it is deeply tied to hacker culture, of which IndieWeb is a clear example, and rhetoric about ‘empowerment’ is common across both types of communities,

instead of ‘users,’ then the field will help to further ‘empower’ and ‘democratize’ society” (2015, p. 11). Further, making cultures engage in knowledge and resource sharing using open-source structures. The combination of open knowledge and resource sharing with material empowerment are described in this literature as helping makers to critique or resist consumer culture. This account presents a multi-faceted definition of empowerment spanning across material and social relations. This breadth of meaning is a rhetorical virtue in that the concept of *empowerment* can bring together actors with varied motivations, such as those who seek personal material empowerment from being able to appropriate technologies to their own ends, and those who want to leverage building to create social change.

I broadly agree that designing for makers rather than users can contribute to empowerment across the multiple dimensions defined in this review. However, like Roedl, S. Bardzell, and J. Bardzell (2015), I believe this potential is in danger of being undermined if HCI researchers and professionals fail to address barriers for access or desire to engage in making, and how they may relate to existing forms of social injustice. In this chapter I focus on people who are engaged in making for the IndieWeb and identify their individual empowerment through this process. Regarding the claim that designing for makers can “democratize society,” it is necessary to identify and take action to prevent exclusion that could undermine that goal. I will discuss this further in Chapter 6, which focuses on influence and exclusion in IndieWeb’s community.

4.2.2 Autonomy and Free Software

Scholarship about Free (Libre) and Open Source Software (FLOSS) communities also provides a valuable foundation for understanding IndieWeb. IndieWeb is largely, though not completely, an example of a FLOSS community. Most IndieWeb projects are released as open source with permissive licenses and IndieWeb’s capacity to work together and grow relies on sharing code and documentation. I acknowledge that IndieWeb is only *largely* FLOSS because there are examples of closed-source IndieWeb software, and these have been embraced by the community. The most visible example is a service called Micro.blog, an IndieWeb-compatible platform for short photos and blog posts, similar to Twitter, but with an option to post to Micro.blog’s community from one’s personal website instead of having your content hosted by Micro.blog directly. A less explicit, but no less important example is that IndieWeb is composed of personal websites that often are not open source. In many cases, some components of personal websites are released under open source licenses, but not the whole site. This is typified in an explanation on IndieWeb’s wiki:

While the entirety of p3k⁴ is not open source, I have open-sourced many of the components. I’ve chosen to provide components of my site rather than the whole thing because in reality nobody is going to want to use the exact same workflow I have built. Instead, components of my site can be re-used by others in their own way. (IndieWeb.org 2020h)

I cite these examples to acknowledge that IndieWeb is not inherently nor wholly FLOSS, but even these exceptions demonstrate a commitment toward free and open sharing—p3k by publishing components as open source, which is framed as being more useful than open sourcing the entire software, and Micro.blog by allowing people to participate via their personal websites.

With that caveat, previous scholarship about values in FLOSS communities provides important background to understand what motivates IndieWeb’s contributors. G. Coleman (2004) asserted the chief

⁴p3k is the software that runs the website of IndieWeb co-founder Aaron Parecki, <http://aaronparecki.com>.

moral goal of FLOSS communities is “a commitment to prevent limiting the freedom of others” (p. 509), a view that is well aligned with IndieWeb’s commitment to ‘autonomy.’ Beyond that, “Political intent and subjectivity are indeed noticeably absent in the constitution of the free software and open source movement” (G. Coleman 2004, p. 508). As with ‘empowerment,’ this type of freedom (a.k.a. autonomy) is presented as politically neutral. This is not to say that all open source communities follow this pattern. Birkinbine (2020) identifies exceptions in his overview about FLOSS and hacker cultures, such as Moglen’s (2003) *dotCommunist Manifesto* and Kleiner’s (2010) *Telekomunist Manifesto*, both of which are informed by radical left politics. Similarly, proposals for *platform cooperativism* (Scholz and Schneider 2016) present concrete proposals for integrating labour politics with hacker culture. These examples highlight a more specific form of political organization than simply sharing code, and political goals extending beyond commitments to ‘freedom’ or ‘autonomy’. There are, of course, individuals in IndieWeb’s community who have articulated strong political views on their websites, during interviews, or elsewhere. However, Coleman’s (2004) account of FLOSS communities demonstrating “political agnosticism” generally applies to IndieWeb when analyzed at a broader level. She describes this ‘political agnosticism’ is shaped by computing practices:

The freedom of free software, while influenced by wider liberal sensibilities, is fundamentally shaped by the pragmatics of programming and the social context of Internet use. My contention is that values for expressive rights as formulated in free software philosophy were and are compelling to programmers because they hold affinities with their technical habitus borne from “practical” (as in meaningful, embodied, and collective action) experiences formed around the pragmatics of programming and the aesthetics of technical architectures. (G. Coleman 2004, p. 511)

This section has identified *autonomy* and *empowerment* as sensitizing concepts for understanding IndieWeb’s values and has explored how those concepts operate in maker and hacker cultures more generally. Both of these concepts are multiplicitous in a manner that engenders interpretive flexibility. This is part of their utility since it helps draw together people with a variety of goals and motivations. In the context of the IndieWeb, these concepts are roughly aligned in a commitment to individual freedom, which is regarded as engendered in part through mastery over technology.

4.3 Principles of building on the IndieWeb

One of the most visible resources for investigating how IndieWeb might employ values in design practices is a wiki article presenting “principles of building on the IndieWeb” (IndieWeb.org 2020i). These principles, listed at <https://indieweb.org/principles> and presented in Table 4.1, are a common entry-point for learning about IndieWeb. Newcomers to IndieWeb’s chat channels are regularly referred to the */principles* page to learn more about IndieWeb. Further, this page was frequently linked in blog posts describing IndieWeb, and its contents were published verbatim by Wired.com (Sterling 2013). And finally, these principles are prominently linked in the sidebar on IndieWeb’s wiki.

Some of IndieWeb’s principles are outcome oriented, by which I mean they refer to goals that are to be achieved as an outcome of building and using an IndieWeb site. Others are used to prescribe processes through which to effectively build on the IndieWeb. This dichotomy is not perfectly neat, since the processes through which individuals build on the IndieWeb are closely linked to self-empowerment

principles

The IndieWeb Community is largely based on **principles** (AKA *tenets*) such as [own your data](#), scratch your own itches, build tools for yourself, [selfdogfood](#), document your stuff, open source your stuff, [UX design](#) is more important than protocols, visible data for humans first and machines second, platform agnostic platforms, [plurality](#) over monoculture, [longevity](#), and remember to have fun!

The IndieWeb community has a [code-of-conduct](#).



Key Principles [\[edit\]](#)

Key principles of building on the indie web, numbered for reference, not necessarily for any kind of priority.

1. 🗑️ **Own your data**. Your content, your metadata, your identity.
2. 🔍 **Use & publish visible data** for humans first, machines second. See also [DRY](#).
3. 🛠️ **Make what you need**. Make tools, templates, etc. for yourself first, not for all of your friends or "everyone". If you design for some hypothetical user, they may not actually exist; if you make for yourself, you actually do exist. Make something that satisfies your needs (also known as [scratch your own itch](#)), and is compatible for others, e.g. by practicing [POSSE](#), you benefit immediately, while staying connected to friends, without having to convince anyone. If and when others join the indieweb, you all benefit.
4. 😊 **Use what you make!** Whatever you build you should actively use. If you aren't depending on it, why should anybody else? We call that [selfdogfooding](#). Personal use helps focus your efforts on building the indieweb around your needs and consistently solving immediate real world problems. AKA eat your own [dogfood](#). [selfdogfooding](#) is also a form of "proof of work 🏗️" to help focus on productive interactions.
5. 📄 **Document** your stuff. You've made a place to speak your mind, use it to document your processes, ideas, designs and code. Help others benefit from your journey, including your future self!
6. 💖 **Open source** your stuff! You don't have to, of course, but if you like the existence of the indie web, making your code open source means other people can get on the indie web quicker and easier.
7. 🏗️ **UX and design is more important** than protocols, formats, data models, schema etc. We focus on UX first, and then as we figure that out we build/develop/subset the absolutely simplest, easiest, and most minimal protocols & formats sufficient to support that UX, and nothing more. AKA UX before plumbing.
8. 🌐 **Modularity. Build platform agnostic platforms**. The more your code is modular and composed of pieces you can swap out, the less dependent you are on a particular device, UI, templating language, API, backend language, storage model, database, platform. Modularity increases the chance that at least *some* of it can and will be re-used, improved, which you can then reincorporate. AKA [building-blocks](#). AKA "small pieces loosely joined".
9. 🗑️ **Longevity**. Build for the [long web](#) 🏗️. If human society is able to preserve ancient papyrus, Victorian photographs and dinosaur bones, we should be able to build web technology that doesn't require us to destroy everything we've done every few years in the name of progress.
10. ✨ **Plurality**. 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.
11. 🎉 **Have fun**. When the web took off in the 90's people began designing personal sites with tools such as GeoCities. These spaces had Java applets, garish green background and seventeen animated GIFs. It may have been ugly and badly coded but it was fun. Keep the web weird and interesting.

Figure 4.1: IndieWeb principles from [indieweb.org/principles](#). Screen captured on January 29, 2020.

as an outcome. However, I present this as a rough scheme for understanding how IndieWeb’s principles present outcomes to be achieved as well as processes through which to achieve those outcomes. Following this scheme, Section 4.3.1 discusses outcome-oriented principles, Section 4.3.2 discusses principles for guiding individuals’ building processes, and Section 4.4 describes processes through which individuals’ activities are coordinated through principles and technical structures to contribute to a cohesive community structure.

4.3.1 Why and what to build on the IndieWeb

In this section, I identify principles that refer to outcomes to be achieved through personal websites with IndieWeb features.

The first principle, *#1: Own Your Data*, reflects a belief that operating one’s own website increases autonomy over one’s data and identity. The nuances of what it means to *own* one’s data are defined differently among members of IndieWeb’s community, although there is a general consensus that owning one’s data means being able to preserve a copy (preferably the original). I asked interview participants how important it was to them to own their data. Ownership was universally considered an essential virtue of IndieWeb’s approach, and 9 of 15 participants included some sort of qualification or definition of this concept as part of their answer. Universally, these definitions acknowledged that owning one’s data can mean different things to different people. Overall, several participants acknowledged that for some people, owning your data means operating your own Web server, but acknowledged that this is an unrealistic goal for most:

Personally, I would prefer if people own their own data and put it all on a server that they have access to under their own domain, but I get that that’s not okay for some people. Like they just don’t have the technical skills or the time to do that. (Interview participant #5)

Ultimately, owning one’s data in the context of the IndieWeb refers to the ability to preserve a copy of that data, and to make sure that the data can remain accessible. In this sense the lowest threshold for ‘owning’ one’s data is *data portability*, the ability to export or transfer personal data to a new server.

I know that some people consider the idea of owning your data [to mean] having it literally live on a hard drive in your house or something. That is not my definition of it. Because I don’t think that’s actually important, I don’t think it’s necessary that everybody should have to have a physical device that they own where the data lives. The point is the ability to choose where it lives and be able to move it to a new place when you want.

(Interview with Aaron Parecki)

Ownership in this context is not about restricting how one’s data is accessed by third parties. In other words, ownership is not about protecting one’s intellectual property, but rather ensuring that retains control and preserves access to a copy of one’s data.

“Own” doesn’t necessarily mean that it doesn’t appear in third-party systems. To some people it does in the community, but the overall definition is that it’s under your control even if it does exist somewhere else. So, a copy of it being under your control.

(Interview participant #12)

To that end, many IndieWeb sites syndicate a copy of their content to social media platforms. For the most part, tools for this syndication are not built by each individual for their own site, but instead most IndieWeb sites that syndicate their content use one of a small number of services to achieve this.

It is commonly asserted that a foundational ingredient for being on the IndieWeb is to own a personal domain name. For example, *Indiewebify.me*, a step-by-step guide for getting on the IndieWeb, asserts that the first step is to “get your own domain name” (*IndieWebify.Me* 2019). This is closely tied to *data portability* because a domain name can be redirected if one moves their online content from one server to another, thus preserving continuity of access. A common pattern among IndieWeb sites among early Web adopters is that they often present a long-standing archive of Web content, preserved at a consistent domain name over time. This was articulated during an interview:

Domains are one of the most reliable digital identifiers you can get. Again, your domain can be taken away in exceptional circumstances generally. You can lose it through accidents, like you forgot to pay your domain provider or something like that. But people have owned domains since the 1990s and still own them. Outside of email addresses at a few really old providers, not many internet identities are that old. (Interview participant #1)

As a result, ownership closely tied to principle #9: *Longevity*, “the goal of keeping your data as future-friendly and future-proof as possible” (IndieWeb.org 2020e). Data portability allows one to retain a copy of data, and the use of personal domain names can facilitate continuity of access to the data at a consistent address.

Another facet of autonomy captured by IndieWeb’s goals is self-expression. IndieWeb’s list of reasons why one should use the IndieWeb (IndieWeb.org 2020l) includes several examples of autonomy that relate to the concept of ownership as discussed previously. In addition, it presents examples of self-expression such as being able to customize one’s site’s visual style and freedom to choose what types of content to publish. This is additionally alluded to in IndieWeb’s principles for building, specifically #3: *Make what you need* and #11: *Have fun*, the latter of which is explained with a reference to customizability of 1990s-era personal websites:

When the web took off in the 90’s people began designing personal sites with tools such as GeoCities. These spaces had Java applets, garish green background and seventeen animated GIFs. It may have been ugly and badly coded but it was fun. Keep the web weird and interesting. (IndieWeb.org 2020i)

Ultimately, self-expression is taken for granted in IndieWeb’s design since it is regarded as an inherent feature of personal websites. In an article published in Slate Magazine and cross-posted to his personal website (in true IndieWeb fashion), journalist Dan Gillmor argued that IndieWeb is important because:

We’re in danger of losing what’s made the Internet the most important medium in history – a decentralized platform where the people at the edges of the networks – that would be you and me – don’t need permission to communicate, create and innovate. (Gillmor 2014b; Gillmor 2014c)

Typical of appeals made in support of the IndieWeb, Gillmor asserts that IndieWeb can recapture a capacity for self-expression that was built into the early Web. In the same article, IndieWeb co-founder Aaron Parecki is quoted describing the early Web, “The original vision was everyone has their own space and made things.” Thus, even though IndieWeb’s principles do not include an explicit call for

self-expression, this is because it is such a core commitment that it is taken for granted as a virtue of owning a personal website.

Commitments to ownership, longevity, and self-expression, as articulated in IndieWeb’s principles, present outcomes to be achieved through one’s IndieWeb site. A large proportion of IndieWeb sites are blogs of some description, often with a large variety of post types including short updates similar to Twitter, photos and videos, long articles, self-tracking (e.g. recording what one eats, reads, watches, or does), location check-ins, replies, likes, and others. Visual styles vary substantially, some sites are sparse and minimalist, others are bright, colourful, and deliberately playful. One site continually changes styles as a visitor scrolls down the page, another dynamically sets the background to a map of the owners’ current location, many use the default visual style of their website software, though may have customized its functionality. An IndieWeb contributor who rarely blogs has used his website as a very detailed one-page profile including his name, age, address, spoken languages, physique, dietary preferences, phobias and other details:

The main thing is just, it’s not a blog but it still embodies me in a recognizable way, I guess. Some people are very recognizable by the content they produce, or by the thoughts that they publish [...] I don’t have a huge need for doing that [...], but I can still be represented quite well by just having some sort of information display like my website does. So, I like to think that at the end of the day, I reach the same level of, after you see my website you know who I am, as some people with blogs do. (Interview participant #2)

The common thread across these sites is that they demonstrate a belief that, by owning one’s own website, one can do whatever one wants with it.

A large number of newcomers to the IndieWeb are attracted because they have prior experiences that affirm these commitments. An interview participant who had his first website in the late 1990s explained that IndieWeb’s principles reflected his early experiences:

The principles. Okay, Aaron and Tantek kind of wrote them down in whenever it was, 2011, but that’s how I came to the Internet. And so, it was a natural kind of fit. (Interview with Jeremy Cherfas)

These principles, therefore, serve as a rallying call to bring together people with similar goals. The next section will describe principles that prescribe a particular set of processes for achieving those goals.

4.3.2 How to build on the IndieWeb

A commitment to building is implicit across IndieWeb’s principles—they are after all principles *of building*. Chapter 1 included a brief discussion about how *building* is a central tenet of the IndieWeb. In fact, this is one of the reasons IndieWeb was selected as a site for this dissertation; its focus on continually building in small increments provides an opportunity to investigate ongoing processes of design and maintenance. This section expands on that earlier discussion about the role of building in the IndieWeb.

The first IndieWebCamp event in 2011 presented a number of requirements for attendance. Among these was a requirement that attendees be “active builders”:

Similar to the Federated Social Web Summit of 2010, IndieWebCamp 2011 is for active

builders only in order to focus the limited time we have on productive real world discussions and code/design/ux sharing that will help us move forward.

Being a builder means you must do one or more of:

- code. create or contribute to indieweb open source projects
- design. create or contribute to indieweb designs, graphic, layout, adaptive or otherwise.
- ux. create wireframes or other indieweb user interface flows

(IndieWeb.org 2019a)

This requirement reflected one of IndieWeb’s founding commitments, to focus on “creators instead of talkers - people that create design/UX/code instead of just email/chat about it” (IndieWeb.org 2015). Since the first IndieWebCamp, this requirement has been relaxed and participants at IndieWeb events need not be builders, but nonetheless, building is a central activity of the IndieWeb. IndieWebCamps, for example, are two-day events, with the first day consisting of talks and discussion sessions and the second day comprised of time to build together (“hack day”).

IndieWeb’s principles include specific guidelines about *how* to build in an IndieWeb way. First and foremost, contributors are encouraged to build for themselves, and use what they make. In fact, when IndieWeb’s */principles* wiki page was first created on September 15, 2012, building for oneself was the founding principle. The entirety of that first revision reads:

Some rough principles of building on the indie web...

1. Eat your own dogfood. Whatever you build should be for yourself. If you aren’t using it, why should anybody else? More importantly, build the indieweb around your needs. Others can do likewise.
2. Add more here.

(<https://indieweb.org/wiki/index.php?title=principles&oldid=1378>)

As of April 2020, the current version of the */principles* page articulates this approach across principles #3. *Make what you need* and #4. *Use what you make*. These are presented as pragmatic guidelines with an assertion that “personal use helps focus your efforts on building the indieweb around your needs and consistently solving immediate real world problems” (IndieWeb.org 2020i).

Closely related, principle #7 asserts that user experience (UX) and design should come before protocols and other lower-level supporting infrastructure: “We focus on UX first, and then as we figure that out we build/develop/subset the absolutely simplest, easiest, and most minimal protocols & formats sufficient to support that UX, and nothing more” (IndieWeb.org 2020i). These three principles emphasize that building should address an observed real-world problem or use-case, beginning with one’s own needs.

During my interview with Çelik, I called him using Slack’s call feature, which neither of us had used before. While we were talking, he wondered if it would be possible to add a link to his website that would direct visitors to contact him directly on Slack. This was a small self-contained task, the main steps of which were to determine how to format a Web link that would open the Slack application and redirect to a specific users’ profile, and then to decide on a visual style for the link. He worked on this feature as we talked, and upon finishing it toward the end of our call, remarked:

Look at that. Through the course of our Q and A we figured out an IndieWeb thing to do and I improved my site. Now I just need to document this on the wiki somewhere. There you go. That’s a real-life example for you.

He framed this example in relation to IndieWeb’s approach more broadly:

That’s core to IndieWeb thinking. Every task we’re doing online, how can we use our own identity and website to accomplish that task, or make the task easier, or make our own websites part of the task flow so that we reinforce the use of our own sites and we reinforce other people using our sites to interact with us online instead of somebody else’s system.

A similar style of building in small increments is evident in the structure of IndieWeb’s in-person events. Two-day IndieWebCamp events conclude with participants presenting brief demos of what they accomplished, which encourages participants to take on small, achievable projects. Homebrew Website Clubs are more frequent—usually biweekly or monthly—two-hour evening meetups run in a variety of cities, in which participants are encouraged to present a demo of recent updates to their site and work on other small tasks such as blog posts. Thus, a large task of customizing a personal website to is divided into incremental units.

This approach nurtures experimentation, wherein new ideas can be tried with minimal risk. Successful ideas can be preserved and iterated upon, while other ideas may remain as small personal experiments or be abandoned. Experimenting on one’s own site also allows people to try ideas that they are comfortable with for themselves but would not impose upon others. One interview participant noted how this approach limited his willingness to share code with others:

My site in many ways has been written in many stressful sprints on Sundays at IndieWeb-Camps. So, the code makes compromises that I’m not willing to make for code that I give to other people. (Interview participant #1)

Another interview participant was in the process of building IndieWeb software that would be deployed for public use, and treated his personal website as a testing ground for that software:

I have to experiment it on my own site a Lot. Like I the other day I lost all my Webmentions.⁵ I’m like, “Huh, what if this happened to somebody?” I’m gonna fix this. (Interview with Jacky Alciné)

This approach reflects a common pattern among IndieWeb’s community of experimenting using one’s own website, which has the effect of mitigating potential harm to others by identifying problems before sharing one’s work more broadly.

This section has identified principles offering prescriptions about individuals’ building practices, specifically around grounding one’s work to observed real-world problems or use-cases, beginning with one’s own needs. While this has described IndieWeb building on an individual scale, the principles and methods described thus far to do not provide a strategy for combining individual websites into a cohesive whole. The next section will discuss how individuals’ building activities are coordinated to contribute to a larger infrastructure.

⁵Webmentions are an IndieWeb protocol for exchanging messages between sites. They are discussed later in this chapter, in Section 4.4.1.

4.4 Coordination mechanisms

This section investigates how a diverse set of individual websites constitutes a cohesive community. Specifically, this chapter has not yet discussed how IndieWeb sites communicate with one another to replicate social media actions. In order to own forms of data such as replies, likes, and other responses that pass between individuals, it is necessary for each site to have a mechanism to communicate with other IndieWeb sites. This is what makes IndieWeb a network of interoperating sites, rather than simply an assortment of independent websites created by like-minded individuals. These mechanisms are built in a decentralized fashion, such that individual sites can operate and control their own mechanisms for exchanging messages and other interactions. This requires coordination among IndieWeb’s community to balance diversity among individual sites with enough consistency to facilitate communication.

The first steps toward this are articulated by IndieWeb’s principles, #5. *Document your stuff* and #6. *Open source your stuff*. Interview participants asserted that documenting one’s work contributes to collective knowledge of successful IndieWeb practices, such as in the following statement:

Doing that whole process of *make what you need, use what you make, document your stuff*, it really forces a lot of insight and self-reflection as well as provides the community with a broader set of perspectives to look at and then be inspired by. (Interview participant #3)

Individuals regularly document their work on their websites, but the main space for documenting IndieWeb’s work is an expansive wiki at <https://indieweb.org>. IndieWeb’s website runs on *MediaWiki* software, which means anyone can edit its articles.⁶ In some cases, IndieWeb contributors focus their attention on using and documenting others’ work:

I decided instead of trying to write my own code [...] I’m gonna spend more of my time trying to popularize this or get other people to use it. And I’m going to use my site as a canonical example of how one could use a site and do these things. [...] I tend to try and focus more on documentation. How do I use this? How could others use this? [...] I can tell from people commenting on my things or people send[ing] me emails that I’m having a much greater impact on the broader community of bringing people in because of things I’ve written or said or done or whatever.

[...]

In some sense I’m going against the some of the more dyed-in-the-wool community basis of you should build it for yourself and write your own code. It’s like okay I can write my own code and I could do this, but is that really the best way to spend my time? I was worried that that would kind of put me as an outsider more than anything else, but at the same time it’s also something that the bigger community needs. (Interview participant #13)

This form of documentation plays a large role in guiding IndieWeb developments. There is a general consensus that new workflows and approaches should build upon existing practices. This is typically framed as a matter of pragmatism, as demonstrated during an interview:

⁶IndieWeb has customized the sign-in mechanism for their MediaWiki so logging in requires Web sign-in (logging into a website using one’s personal Web address). Thus while anyone can edit IndieWeb’s wiki, in practice this is limited to people who have a personal website and have configured it to support this feature, which demonstrates at least some agreement with IndieWeb’s approach.

We try to be very pragmatic about stuff in the IndieWeb community. So, in the end it's just, what choice is the easiest to implement or what are people already building upon? [...] We don't want theory to win out over practicality, basically. I think that's the usual way stuff gets sorted out in the IndieWeb. Who is already publishing this data or who is already doing XYZ and what does it look like? Does it work for them? If it works for them, are there any reasons that it wouldn't work for me? And if you can't really come up with a good reason, and someone else is already successfully using a pattern, then that's the one you write down and settle on. (Interview participant #2)

Similarly to how documentation contributes to community knowledge, open sourcing one's work contributes to community material resources in the form of code. When code is released under an open source license, it is possible for others to build upon it by (a) copying code to their own projects and/or (b) submitting changes in the form of pull requests. Pull requests are a mechanism by which, once a developer has “forked” a copy of an open source code repository and made changes to that copy, she or he can request that those changes be *pulled* into the original repository by its maintainer. This is one of the mechanisms through which individual IndieWeb projects contribute to a larger system.

In these practices of knowledge and resource sharing, it becomes evident that IndieWeb's values extend beyond the self. Building what one needs, and then sharing knowledge and resources that result from that endeavour, demonstrates a commitment to benevolence. Here I draw from Schwartz' definition, in which he asserts that the driving goal of benevolence to be “preserving and enhancing the welfare of those with whom one is in frequent personal contact (the ‘in-group’)” (S. Schwartz 2012, p. 7). He makes two additional observations about benevolence that are helpful here. First, that benevolence and power—the pursuit of “social status and prestige, control or dominance over people and resources” (2012, p. 5)—are in conflict with one another. Much of IndieWeb's *raison d'être* is its opposition to concentrations of power among platforms that restrict access to resources. Second, he differentiates benevolence from universalism, which is characterized by concern for the “welfare of all people and for nature” (2012, p. 7). This is not to say that IndieWeb's contributors lack such a concern—in fact Chapter 6 includes a discussion about IndieWeb's efforts to be inclusive—however, the practices of documentation and sharing open source code are predominantly focused on an *in-group* insofar as they support the growth IndieWeb's community around shared practices.

Von Krogh et al. (2012) reviewed ten years of research about motivations for contributing to open source projects, and this review provides useful insights for understanding how FLOSS structures can support collaborative and benevolent modes of work. They summarize that past research has generally classified motivations of FLOSS developers as intrinsic (ideology, altruism, kinship, fun), internalized extrinsic (reputation, reciprocity, learning, own-use), and extrinsic (career, pay). They argue, however, that “[open source software] development differs from conventional software development along three dimensions (incentives, control, and coordination mechanisms), which in turn are reflective of a distinct social practice in which ethics plays a central role” and that instead of investigating individuals' motivations in isolation, “scholars should expand theory building and research to cover the interplay with institutions, goods, and the social practice” (2012, p. 670). In developing a research perspective that moves beyond individuals' motivations in this manner, they highlight research that has identified code itself as a coordination mechanism:

MacKenzie (2005) suggests that the OSS code itself, with modular, functional, and transparent objects, gives rise to a social practice with its own ethics. The technical object of the

software code requires developers to behave in a specific way when creating and maintaining it, for example, modularizing, reusing, keeping to the API specification, or taking great care to document (Baldwin and Clark 2006). (von Krogh et al. 2012, p. 670)

This argument is consistent with Coleman’s (2004) assertion that FLOSS ideologies extend from technical practice, rather than the other way around. Kelty (2008) expands this argument with his description of FLOSS as demonstrating a *recursive public*, which he defines as:

A public that is 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; it is a collective independent of other forms of constituted power and is capable of speaking to existing forms of power through the production of actually existing alternatives. (Kelty 2008, p. 3)

The independence of a recursive public is not absolute, but instead “provisional and structured in response to the historically constituted layering of power and control within the infrastructures of computing and communication” (2008, p. 9). Recursion arises from such publics continually producing and reproducing the infrastructure that enables them. IndieWeb’s principles presume building itself to be a foundational activity, and then advocate specific approaches to building intended to support a community infrastructure. This is exemplified in a claim by Çelik that “we’re gonna use the web itself to build the web that we want” (Çelik 2014).

I assert then, that IndieWeb is a clear example of a recursive public. In the remainder of this section, I will discuss how material features of IndieWeb’s technical standards and other components—its “building blocks”—operate as coordination mechanisms that convey IndieWeb’s values, and simultaneously contribute to the community’s understanding of those values and their possibilities.

4.4.1 Technical standards / Building blocks

Technical standards are the foremost material structure through which individual websites are made commensurable. Standards enable interoperability by defining shared concepts and practices. In the context of the IndieWeb, even as two websites may differ substantially in structure, appearance, and content, they can communicate with one another as long as they have a shared understanding of what an act of communication entails.

Insofar as IndieWeb is intended to be flexible, there is no unified IndieWeb standard, but instead a collection of small standards which may be implemented piecemeal. A partial list of IndieWeb standards is reproduced in Table 4.4.1. Of the standards listed, Webmention, Micropub, and WebSub have been published as recommendations by the W3C. They were produced by the Social Web Working Group, which operated from 2014-07-21 to 2018-02-13 with a mandate to “define the technical standards and APIs to facilitate access to social functionality as part of the Open Web Platform” (W3C 2014). This means that these standards have been reviewed by W3C members, software developers, other W3C groups, and interested third parties, and are recommended for widespread deployment (Parecki 2017). The remaining standards have not gone through the W3C’s review process and are instead published by IndieWeb directly.

Beyond the summary in Table 4.4.1, this dissertation does not endeavour to present a detailed description of each of these standards. Instead, it presents two key standards which are necessary for

Table 4.1: Partial list of IndieWeb technical standards with definitions.

Name	Description
IndieAuth	IndieAuth defines a protocol allowing people to sign-in to third-party services using their website as their identity. This works similarly to sites that allow you to sign-in using a Google, Facebook, Apple, or other account, but without transmitting data through those intermediaries.
Microformats 2	Microformats defines language for marking up people, events, blog posts, and other entities using HTML. Microformats 2 is the current iteration of this specification and is widely used in the IndieWeb to make websites machine-readable.
Micropub	“The Micropub Standard defines a protocol used to create, update and delete posts on one’s own domain using third-party clients” (IndieWeb.org 2018c). In practice, this spec allows people to update their website using a variety of mobile and web applications, several of which include quick actions for liking or replying to posts in their newsfeed.
Microsub	The Microsub specification defines a standardized way for building “feed-readers” that are used to follow posts from a variety of websites and social media - similar to one’s timeline on Twitter or Facebook. Microsub divides feed-readers into two halves: “A Microsub server manages the list of people you’re following and collects their posts, and a Microsub app shows the posts to the user by fetching them from the server” (IndieWeb.org 2020g).
Webmention	Webmention defines a method to notify a URL when it is linked from one’s website. In combination with Microformats 2, Webmentions are used to send replies, likes, and other interactions among web posts.
WebSub	WebSub defines a mechanism through which publishers of web content can send updates to their subscribers when new content is available. This defines a relationship similar to following someone on popular social media platforms.

understanding how the IndieWeb functions, *Webmention* and *Microformats 2* (MF2). In combination, these standards allow a wide variety of interactions to be communicated between individual websites, such as replies, likes, shares and other actions familiar on social media platforms.

Webmention defines a method for notifying a URL when it is linked from one’s website. To send a Webmention, one simply links to another site in the content of a blog post or other Web content. Then, one configures their website or delegates a third-party to send a Webmention consisting of two properties, a *source* indicating the URL from which the Webmention originates, and a *target* indicating the URL to which the source is linking. As described in the Webmention spec Parecki 2017, if Bob uses his website to write a reply to Alice, he can send a Webmention with the following content:

```
source=http://www.bob.example/post-by-bob
target=http://www.alice.example/post-by-alice
```

When sending a Webmention, Bob’s website or the third-party service he is using to send Webmentions will first check search for a link to a *Webmention* in Alice’s HTML. This is typically indicated through a line of HTML like:

```
<link href="http://alice.example/webmention-endpoint" rel="webmention" />7
```

A Webmention endpoint defines the URL to which a Webmention is sent. It may be hosted on one’s own server, or by a third party, and is responsible for validating that the Webmention is correctly formatted, that it is authentic—i.e. that the *source* URL actually contains a link to the *target* URL. In most cases, once they have been received, Webmentions are published by the receiver, typically as a comment or other response to the *target* post. For example, Alice might configure her site to display a comment below her post such as “This post was linked by <http://www.bob.example/post-by-bob>.”

Webmentions are an exemplar of IndieWeb’s efforts to define *simple* standards. They fulfill the basic task of sending a mention from one URL to another, and nothing else. Conventions about if and how to display received Webmentions are defined elsewhere. In the case of Webmentions, simply displaying that some other URL linked to one’s post is not very meaningful, since readers would have to visit that URL and read its content to see *what* that link actually conveyed. To resolve this, community consensus among IndieWeb sites involves parsing the contents of the *source* URL using MF2.

MF2 defines a vocabulary for marking up people, blog posts, and other entities in HTML. Combined with Webmentions, they can be used to discern contextual information about Web content. MF2 works by adding a *class* to an HTML element indicating its type. For example, an individual entry (e.g. an article or a blog post) is defined by adding a class called **h-entry** as in the following example:

```
<div class="h-entry"> [article goes here] </div>
```

Additional information is encoded using a variety of other classes, such as **p-name** to indicate the name or title of an article, **dt-published** to indicate the date and time at which the entry was published, and **h-card** to represent people or organizations (similar to a business card). Parsing this content entails translating the marked-up HTML into a hierarchy of nested elements. For example, an **h-feed** may contain several **h-entry** elements, each containing an **h-card** element to describe its author, which in turn may contain properties describing the author’s name, profile photo, or other information.

⁷This follows a convention of using HTML’s `<link>` element to identify an external resource, denoted by the `href` property. The `rel` property defines the relationship of this resource to the page, with a value of “Webmention” indicating that the resource is a Webmention endpoint.

In the example of Bob and Alice exchanging a Webmention, Bob can use MF2 to identify which parts of its HTML represent information about himself, such as his name and profile photo, and information about the post itself such as its publication date, type (e.g. article, reply, like, RSVP), and content. If Alice’s Webmention endpoint is properly configured, it can parse this information from Bob’s post, and describe the contents of the *source* URL. Then Alice’s website could display Bob’s Webmention in more detail, such as “Bob liked this” or “Bob replied: Great post, Alice!”

The Webmention specification notes that recipients of Webmentions may parse the *source* website and display the results, but does not prescribe a method for doing so:

The receiver MAY publish content from the source page on the target page or other pages, along with any other data it picks up from the source. For example, the receiver may display the contents of the source as a comment on the post, or may display the author’s profile photo in a list of others who have sent similar Webmentions, e.g. showing a list of people who have all “liked” a post. (Parecki 2017)

As a result, individuals or organizations implementing the Webmention specification can choose if and how to display such content. A convention has emerged around the MF2 approach described here, but individuals may choose to ignore this convention or even attempt to establish a new practice that could be adopted by others. This is typical of IndieWeb’s “building block” approach, which IndieWeb’s wiki connects to Berners-Lee’s principles for the Web itself:

Why building blocks? This early quote from Tim Berners-Lee’s Weaving The Web Chapter 4 page 39 summed it up well:

... if I had insisted everyone use HTTP, this would also have been against the principle of minimal constraint. If the Web were to be universal, it should be as unconstraining as possible. Unlike the NeXT computer, the Web would come as a set of ideas that could be adopted individually in combination with existing or future parts.

As was the Web, so is the IndieWeb.

The IndieWeb also comes as a set of (additional) ideas (as building blocks), that can be “adopted individually in combination with existing or future parts”.

It is for these reasons that the IndieWeb is built with a set of building blocks, instead of on a monolithic “stack”. (IndieWeb.org 2019c)

This approach articulates IndieWeb’s principles #8. *Modularity* and #10. *Plurality*. IndieWeb’s standards are modular because they are focused and can be adopted in a variety of combinations. This in turn supports plurality because it is possible and encouraged to create multiple implementations of IndieWeb’s standards. For example, as of April 21, 2020 IndieWeb’s wiki lists 25 examples of software and 11 services that can be used to publish Webmentions from one’s website (IndieWeb.org 2020k). The same is so for almost all IndieWeb building blocks; there are multiple ways to do things and this is celebrated.

Modularity of this sort makes development easier in several respects, since it allows people working on IndieWeb projects to incorporate ideas and code or even entire ‘modules’ from other people’s projects. It contributes to plurality by making it easier for individuals to customize their IndieWeb site. Plurality,

in turn, create challenges as well. First and foremost, it can be difficult for newcomers to understand how to use IndieWeb because there are always multiple valid approaches. Similarly, for those who want to develop for IndieWeb, it can be difficult to investigate implementations in code. One individual wrote in chat that it could be awkward navigating IndieWeb source code because it is spread across individuals’ largely unpolished GitHub repositories. Çelik defended IndieWeb’s approach by stating, “variety and diversity is better than /monoculture in the long run, always”⁸, and asserting that success in the IndieWeb is not measured by creating an app or platform with a large user-base, but instead “success is achieved if even you yourself use your software regularly for your primary web identity.”⁹

An entry on IndieWeb’s wiki frames plurality as a defining feature of IndieWeb’s approach:

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 IndieWebCamp¹⁰ different (more inclusive) than all other such “open source” efforts out there. (IndieWeb.org 2018b)

I will briefly discuss two features from this passage. First, the claim that plurality makes IndieWeb more “inclusive” refers to the ease and freedom of developing IndieWeb sites and tools. Since individuals are encouraged to build for themselves, and can adopt small standards piecemeal, it can be quite easy start building for the IndieWeb compared to other types of social networks:

But I still like this approach of being open towards other things both in integration to other networks and [...] the philosophical approach to relatively easy standards. Many independent implementations. For instance, it’s way easier to get a basic website with a few IndieWeb building blocks going from scratch than something like Mastodon integration.

(Interview participant #1)

The comparison to Mastodon is worth a brief digression. Mastodon is a decentralized social network first launched in 2016. Its design demonstrates many philosophical similarities to that of IndieWeb, and its technical foundation, *ActivityPub*, was published by the W3C’s Social Web Working Group that also published IndieWeb’s *Webmention*, *Micropub*, and *WebSub* specifications. Joining Mastodon as a user is a simple matter of signing up for an account. However, building Mastodon support into one’s own website is significantly more complicated than adding MF2, Webmention support, or other basic IndieWeb features. Compared to IndieWeb, Mastodon’s approach is far more conventional in that it can be understood as a stable, singular piece of software. By contrast, that each IndieWeb site may be running entirely different software and may adopt individual IndieWeb building blocks in various combinations can make IndieWeb conceptually difficult to grasp. Yet, this heterogeneity makes actually building for the IndieWeb simpler compared to other decentralized Web projects.

Second, the claim that IndieWeb “figure[s] out how to interoperate between different coding/implementation approaches” after people have built “what works for them” presents an opportunity to loop back to IndieWeb’s commitment to *building for yourself*. This section has argued that IndieWeb’s technical standards operate as coordination mechanisms because they establish structures for shared intelligibility among a plurality of diverse implementations. In the example presented here, regardless of

⁸‘/monoculture’ refers to <https://indieweb.org/monoculture>

⁹<https://chat.indieweb.org/meta/2018-09-05#t1536189771619000>

¹⁰Since IndieWeb began as an event called “IndieWebCamp,” there are places in its documentation where the older term is used to refer to the IndieWeb more generally.

differences among IndieWeb sites, those that implement Webmentions and MF2 can communicate with one another in a standard fashion. If implementations come before interoperability, then it is important to note that standards would not be imposed from the top down.

Aaron Parecki, who has worked on several IndieWeb standards, explained that although he has encountered people who believe specification writing to be a top-down process, his approach focuses on capturing existing ideas and practices:

I hear a lot of people who think that spec development is like someone comes up with an idea and then invents a thing and that becomes a spec. While that does actually happen, and there are plenty of examples of where that has happened both successfully and not successfully, I think that’s not actually a great example of being a good spec writer, because that’s essentially benevolent dictatorship.

So the way that I approach it, which is I think something that I was introduced to in a formal way throughout the W3C Process, is that the role of the editor is—it’s not that you are inventing something and then saying hey everybody go do this. It’s reconciling multiple viewpoints, combining them, figuring out what to include, what to not include.

(Interview with Aaron Parecki)

In practice, IndieWeb’s standards do involve acts of invention. Parecki acknowledged that he began writing an IndieWeb specification called *Microsub*¹¹ “from thin air essentially.” He expressed that he could not know whether the spec was successful until seeing it implemented in software both by himself and by others:

In the time between when I sort of finished my first draft of it and started actually building something, I started getting really nervous because I don’t like the idea of a spec existing without anything built with it. That is not what a spec is supposed to be. A spec is supposed to be capturing best practices around this concept, not just a document that says, “This is how it is.”

(Interview with Aaron Parecki)

Even in this case, where the initial Microsub specification was akin to a top-down declaration, subsequent developments have been shaped by implementations. Nonetheless, in the following sections I show that the initial top-down standardization from the Microsub specification is a defining feature.

This section has discussed principles, practices, and technical components that support individual websites and software to operate as part of IndieWeb’s broader infrastructure. Documentation and open source practices reflect a commitment to benevolence, which helps various members of IndieWeb’s community achieve autonomy and self-empowerment by building upon each other’s work. IndieWeb’s standards and building blocks coordinate individual projects into a collective enterprise while attempting to preserve individual virtues through a commitment to plurality.

4.5 Building Yarns

In this section I describe my own processes of building software based on IndieWeb’s building blocks, called. Following tenets of a critical making approach (Ratto 2011), I focused on how the processes of making could help me better understand how IndieWeb’s sociotechnical infrastructure related to my

¹¹Microsub is a specification for building social reader applications, which will be described in Section 4.5

conceptual framework. This approach is described in more detail in Chapter 3, Section 3.2.1. To recount this process, I draw from field notes, logged interactions on GitHub (where Yarns’s source code is stored), and in some cases memories of the process, i.e. *headnotes*. Where headnotes are used, they are supported by field notes—e.g. headnotes have been useful for reflecting upon connections among concepts, which have then been validated through further observation, coding, and reviewing field notes. In some cases, I use interviews and other materials to explain how my own building activities were influenced by other parts of the IndieWeb, often unbeknownst to me at the time I was building.

The software I built is an IndieWeb-friendly feed reader called *Yarns*. It is used to follow and interact with feeds from a variety of websites, similar to one’s news feed on platforms such as Facebook and Twitter. It is installed as a plugin for websites running WordPress, a popular open source content management system (CMS). Yarns itself was developed in two iterations. The first, which was called *Yarns Indie Reader* built upon an existing IndieWeb reader plugin for WordPress called *WhisperFollow* (McAllan 2014). After this first stage of development, I learned about an emerging IndieWeb specification for building readers, called *Microsub*. This prompted a second iteration that utilized this specification, titled *Yarns Microsub Server*.

Through both phases of development, Yarns had to interoperate with other areas of the IndieWeb. Subscribing to diverse websites and interoperating with other IndieWeb applications made the process of building Yarns relatively complex. During a talk at an IndieWebCamp in 2019, one of IndieWeb’s co-founders noted that IndieWeb readers are not necessarily something everyone can build for themselves:

The challenge is that it’s actually kind of a tough problem to create a service that manages subscriptions and does all the feed reading and it is not necessarily practical to have everybody build their own into their own website, which means it does make sense to have these services to manage this. So how do we reconcile those two views of we don’t want everybody to have to build their own feed subscription management stuff, and yet we also want you to be in control of it and have that data somewhere that you choose? (Parecki 2019)

It was precisely this challenge that made building Yarns be a useful endeavour for understanding how individual IndieWeb projects fit into its larger system.

4.5.1 Motivations and background

When I decided to build IndieWeb software, I took seriously IndieWeb’s principles of “making what you need” and “using what you make.” While attending my first Indie Web event (IndieWeb Summit 2017), I reflected on how my needs and wants differed from what I’d seen many others working on. Many of the IndieWeb projects I saw were focused on publishing various types of Web content to personal websites, including things like detailed self-tracking. By contrast, I was more interested in reading and responding to what others were posting on their IndieWeb sites. Essentially, I wanted to replace my Twitter timeline with a timeline of IndieWeb posts. At the time I was brainstorming project ideas, there were a few options for doing so, but none met all of my criteria, which were:

- Self-hosted. I wanted to host my social timeline on my own server. This resonated with IndieWeb’s “Own your data” principle. Specifically, I wanted to control how my timeline is sorted, and ensure the timeline’s functioning would not change without my knowledge and consent.
- Easy to install and use. I wanted to build something that would be accessible to people with varying levels of technical skill.

- Ability to respond to what I’m reading with standard social media actions, such as replying to or liking posts. When user responds to a feed item, the response should be published to their website as a post. If the user has implemented Webmentions on their site—for example by installing the Webmention plugin created by IndieWeb contributors—then a Webmention will be sent to the post to which they are replying.

While those formed a baseline, I also wanted to include the following feature if possible:

- Be able to follow people on popular social media platforms like Twitter.

As well as being practically useful, this could help me understand what’s involved in building IndieWeb systems that interoperate with corporate Web platforms.

With these goals in mind, I reviewed existing IndieWeb solutions. Two readers that had previously been built by members of IndieWeb’s community offered some of the features I wanted. The first of these readers was a WordPress plugin called *WhisperFollow* (McAllan 2014). *WhisperFollow* fulfilled many of the requirements listed above. It runs on one’s own server (as long as one is running WordPress), it is easy to install (WordPress contains simple mechanisms for installing plugins with which most people who operate WordPress sites are familiar), and it can parse content from many IndieWeb and other websites. However, at the time I began working on yarns, *WhisperFollow* did not include features for replying to or liking posts, except for “reblogging,” which posts a copy of a post to one’s blog (equivalent to retweeting).¹² Another IndieWeb reader software I looked into was *Woodwind* (Mahan 2017). This software includes several functions not available in *WhisperFollow*, such as the ability to reply to one’s feed with *likes* and *replies*, as well as a better interface for subscribing to feeds. However, *Woodwind* did not meet my requirements for being both easy-to-use and allowing the user to host their data on their own server. The easiest way to access *Woodwind* was by creating an account at *woodwind.xyz*, a publicly accessible version that is no longer running¹³. While it was operating, *woodwind.xyz* was simple to access, but experienced occasional service outages that prevented users from accessing their feeds. It is possible to install *Woodwind* to one’s own server, which requires one to know how to set up a server running Python, create database tables, and run a program using a command-line terminal. I perceived that knowledge requirements as a barrier for people who are not professional or hobbyist developers. Based on these observations, neither *WhisperFollow* nor *Woodwind* quite met my requirements individually, but both were strong influences on my design for Yarns.

4.5.2 From *Yarns Indie Reader* to *Yarns Microsub Server*

The first version of Yarns was built by adapting *WhisperFollow*’s code. I opted to develop a WordPress plugin because my personal website uses WordPress and I have developed for WordPress in the past. Because *WhisperFollow* is a WordPress plugin that includes many features I wanted to include in Yarns, it was possible to copy and adapt its code, rather than starting from scratch. Additionally, *WhisperFollow* was released under the GNU General Public License version 2, which permits copying, distribution, and modification as long as the derivative is released under the same license (*GNU GPLv2* 1991).

The first set of hurdles was developing familiarity with relevant WordPress functions and data structures. I had experience building WordPress themes¹⁴ in the past, but Yarns required me to learn several

¹²Additional ways to interact with one’s feed were added in 2018, but were not available when I began working on Yarns.

¹³According to IndieWeb’s wiki,

¹⁴WordPress, like many CMS’s, allows users to customize the appearance and structure of their website using themes.

features with which I was unfamiliar. Notably, I learned about WordPress’s database structures to store newsfeed data, and about scheduling recurring tasks—e.g. regularly checking one’s subscriptions for new posts—with WordPress’s *cron* functions. To learn these steps, I relied heavily on the WordPress Codex,¹⁵ an online manual including detailed documentation for developers. Additionally, I carefully read WhisperFollow’s code, since in most cases I was modifying or building upon its functions rather than creating something new.

By February 2018, I had completed a first version of *Yarns Indie Reader*, which I announced in a blog post at <https://jackjamieson.net/yarns-indie-reader/>. Figure 4.2 shows a screenshot of Yarns from this time. The screenshot illustrates how a post is displayed in Yarns’s timeline, along with buttons for replying to or liking the post.

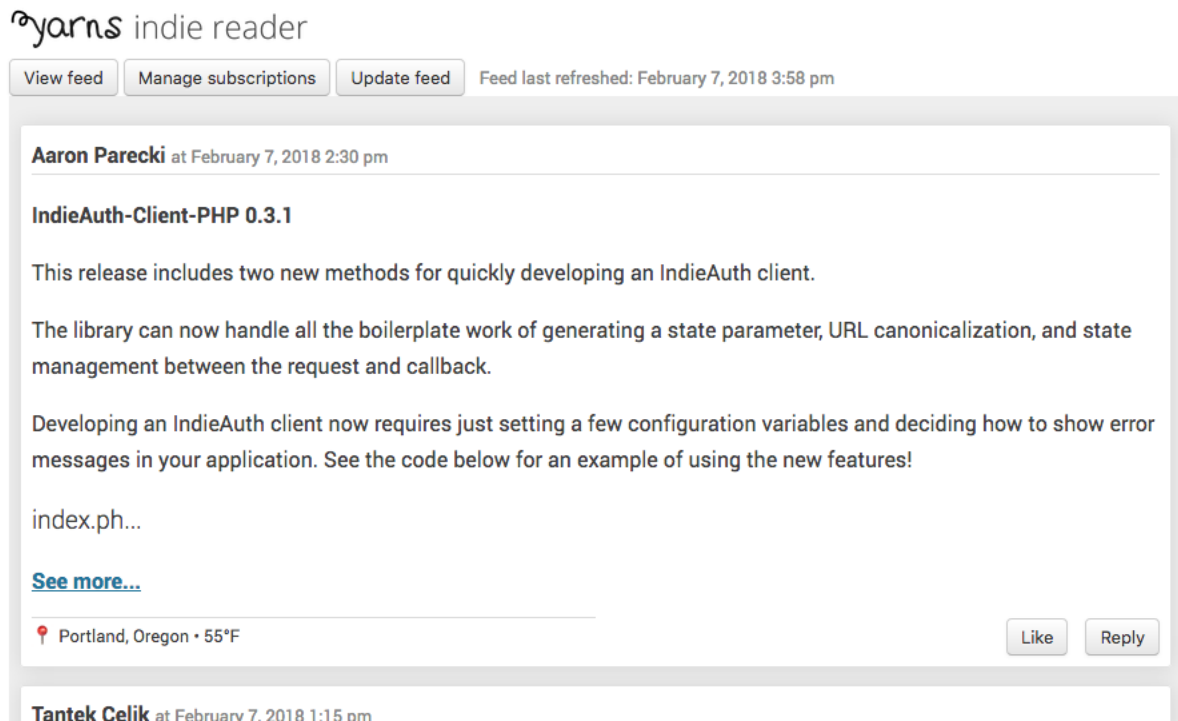


Figure 4.2: Screenshot of *Yarns Indie Reader*.

Much of my development from September 2017 through February 2018 was solitary. In part this is because I viewed the early stages of my data collection as an opportunity to focus on observation, so as to develop technical fluency to better understand IndieWeb discussions and to prepare myself for conducting interviews. To those ends, this initial work on Yarns was useful. However, my observations became much richer as I became more actively involved with IndieWeb’s community around June 2018.

Around the same time as I had been working on Yarns, others in the IndieWeb community had been developing a different approach to readers called *Microsub*. This provided an opportunity to improve the design of Yarns, and more importantly to integrate it more closely with other IndieWeb projects. Microsub is a technical specification that divides reader software into two components. As stated in the spec document, “The Microsub server is responsible for managing the accounts you follow, retrieving

Themes essentially consist of a collection of templates for various sections of a website, and in many cases additional code to add functions beyond WordPress’s default capabilities.

¹⁵<https://codex.wordpress.org/>

updates from them, and the Microsub endpoint provides the feed entries in a normalized format for easy consumption by clients” (IndieWeb.org 2019e). Rather than providing a user interface for viewing this content, the server defines an API. A Microsub client connects to that API, making requests for specific pieces of content (such as a list of feeds, or an array of feed content). In essence, this means the server can operate entirely in the background and, apart from an initial setup process, users can interact solely through an interface provided by a Microsub client. Figure 4.3 shows Yarns’s interface for organizing feeds in channels next to a Microsub client. Because I had built Yarns Indie Reader without this spec in mind, I had worked on both halves at once. A large portion of my development time had focused on how feeds were presented to a user and the results were mediocre. When it comes to building for the Web, I’m better at the back-end work involved in collecting and standardizing data than I am at designing user interfaces. (In fact, standardizing and cleaning data for Yarns draws on the same skillset used in this dissertation’s logged data analysis, reported in Chapter 5). Thus, I could immediately see the benefit of Microsub’s separation of server from client. In addition to allowing me to offload the work of designing a user-interface, this approach makes it possible to use multiple clients for different purposes, such as a browser-based client for viewing feeds at home, and a mobile app for viewing feeds on one’s phone. Most gratifyingly, this shows that other members of IndieWeb’s community shared my view that readers were at that time the most important gap in IndieWeb’s material infrastructure. To some extent, I felt a little foolish that I hadn’t caught on to the emergence of Microsub earlier, but at least I had been on the right track.

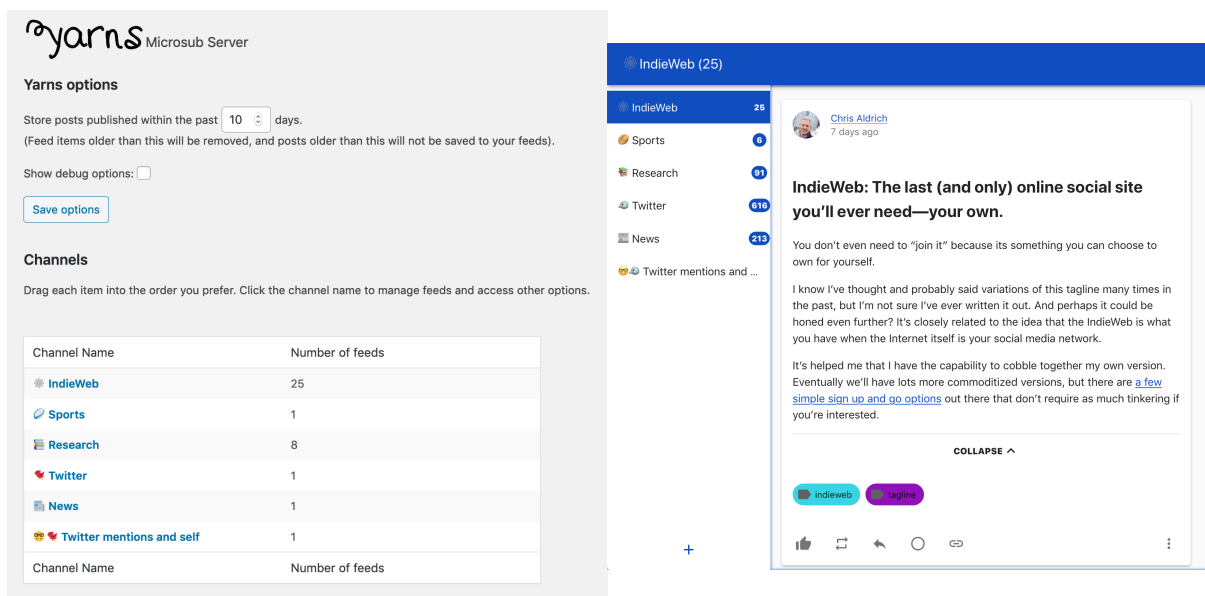


Figure 4.3: Left: Screenshot of Yarns Microsub Server interface for organizing subscriptions into channels. Right: Viewing feeds in a Microsub client called Together (<https://alltogethernow.io>)

I attended IndieWeb Summit, the largest annual IndieWeb event, from June 26-27, 2018 in Portland Oregon USA. About a week before this event, I published a blog post mulling over the future of Yarns, and whether to adapt it to make use of the new Microsub spec. I received comments from a few IndieWeb contributors encouraging me to consider adapting Yarns into a Microsub server. At that time there were a few Microsub clients that had recently become available, and two servers, neither of which were easily accessible to the public. Multiple implementations of Microsub clients and servers alike would

serve IndieWeb’s commitment to plurality, and help validate the quality of the spec. The editor of the Microsub spec explained that writing the spec, and even implementing it himself was only

“half the story because just because I can build something based on the spec doesn’t mean it’s actually a good spec. I need to be able to show that other people are also able to read it and understand what to do, and then build functional software” (Interview with Aaron Parecki). He further commented on the role of multiple Microsub projects:

Let’s try to get people to build stuff now and see if this is actually working before we go any further with it. So that’s when we started seeing the iPhone App, the Android app, the other servers being developed, the JavaScript *Together* app, and all those started getting developed in this framework. And it turns out that it is working and it has actually helped the way I thought it would help, which is that each side of this picture can iterate faster and make more progress because they don’t have to build both sides of this thing. So, it’s been really reassuring to see all this progress being made on it, because it is demonstrating that the spec is drawing the right lines between the two different sides of things and it is written well enough to instruct somebody how to build something. (Interview with Aaron Parecki¹⁶)

Yarns was in a suitable state to be modified into a Microsub server because it already included functions for subscribing to and parsing content from other websites. I saw this as an opportunity to improve upon Yarns and to better understand the IndieWeb by participating in an emerging area of development.

IndieWeb summit follows the same two-day format as IndieWebCamp events as described earlier in this chapter. The first day is dedicated to presentations and discussion sessions, the latter of which are proposed by attendees the day of. During the second day, attendees undertake development or design projects of their choosing. I arrived in Portland with a plan to get started on Microsub during the event. The first day began with a series of keynote talks, including several comments about readers. Çelik (2018) presented a talk about the *state of the IndieWeb*, in which he suggested that the coming year would be “the year of the IndieWeb reader:”

Ever since we lost Google Reader¹⁷, we’ve had many other small efforts at kind of recreating old RSS reader UIs, but not really getting in terms of the modern social Web. In terms of what people expect, they expect a mobile client, they expect a client that they can interact with, that they can quickly favourite or like or retweet posts. [...]

The IndieWeb aspect of this is all those interactions, instead of going to a social media silo, get sent from the reader to your own website. So, when you’re done reading you’ve got a whole record of the things that you’ve replied to, liked, or whatever, on your own website. (Çelik 2018)

Related to Çelik’s comment, Parecki (2018) presented a keynote in which he explained how Microsub worked, and announced that his Microsub server, *Aperture*, had just gone into public beta. I began working on a Microsub version of Yarns, dubbed *Yarns Microsub Server*, during the second day of IndieWeb Summit. The first step was to implement an *API endpoint*, which is essentially a gateway through which Yarns would communicate with a Microsub client. Not having built something like that before, I copied source code from the *Webmention* plugin for WordPress, which included its own endpoint,

¹⁶Note: This interview was conducted after several Microsub servers had begun development, including Yarns.

¹⁷Google reader was a popular service for subscribing to RSS and other feeds, which was discontinued in 2013.

and attempted to modify it for my purposes. (Later, I redid this by basing my code on the *Micropub* endpoint, which more closely mirrored the capabilities necessary to implement *Microsub*). By the end of the day, I had made limited visible progress—attempting to access Yarns’s endpoint consistently resulted in an error message—but was beginning to understand how Microsub worked. Much of the conceptual progress I made was a benefit of attending IndieWeb Summit in person, since I was able to talk and troubleshoot with other attendees who were working on Microsub projects of their own, or had experience building similar systems.

4.5.3 Plurality in IndieWeb sites

The component of Yarns that has taken the longest time to implement and maintain is parsing websites. IndieWeb’s commitment to *plurality* is one of the reasons for this difficulty, since each IndieWeb site can be structured differently. Because the first version Yarns was adapted from WhisperFollow, it could build upon WhisperFollow’s parsing functions, which worked with websites whose content is structured using MF2 as well as sites that publish feeds using RSS or the Atom Syndication Format.¹⁸

MF2 feeds are the most useful for following other IndieWeb pages because they present more detailed information about feed contents. A key difference between RSS/Atom and MF2 is that MF2 includes properties for describing social Web actions, such as recognizing that a feed item represents a *like* or *reply* to another post. Encoding this information in a machine-readable structure makes it possible for Yarns to represent such actions in a standardized form, much like they are presented in popular social media. However, MF2 feeds are also more complex to parse than RSS and Atom, since MF2 is a more flexible standard. Specifically, RSS and Atom feeds are presented in .RSS or .XML files, which are not intended to be read by humans directly, and MF2 feeds are constructed directly in the same HTML that is viewed by human readers of a website. One consequence of this is that MF2 Feeds themselves can be structured in various ways, some of which are illustrated in Table 4.2. The first row in this table describes the most conventional structure, in which entries are identifiable because they are contained within an explicitly denoted **h-feed**. The second row describes a common situation in which entries were not placed within an **h-feed** and so must be identified directly. The third row describes a rare case that posed two particularities. First, **h-feed** elements are contained within an **h-card**. Second, multiple feeds are present—in this case to organize the author’s entries by type (such as distinguishing articles and events). As a result of varying feed structures as exemplified in this table, as well as other differences among IndieWeb sites, the code I wrote had to be continually updated as new cases were identified (almost always as a result of breakdowns where parsing failed).

Moreover, MF2 is constantly iterated upon to incorporate new post types and other properties, or to reflect cases where actual use deviates from the standard. For example, MF2 includes notation for describing events, including the event’s name, start and end times, location, description, and other properties. According to the MF2 specification, event descriptions were to be identified with the **p-description** property, but in 2018, editors of the MF2 standard identified that most people were using the **e-content** property instead. The prefixes for these properties indicate how they should be parsed, with **p-** indicating that **p-description** is *plain-text*, so any HTML in this field should be stripped when parsing. By contrast, **e-** indicates that **e-content** is an *element-tree*, meaning that it can contain HTML elements such as images, links, and an organizational hierarchy of elements, the HTML

¹⁸RSS and Atom are competing feed formats that present website contents in a standardized, machine-readable way.

Table 4.2: Multiple structures for structuring feeds in MF2.

Type	HTML (minimal example)	Parsed structure
h-feed with entries	<pre><div class="h-feed"> <div class="h-entry"></div> <div class="h-entry"></div> <div class="h-entry"></div> </div></pre>	<ul style="list-style-type: none"> • h-feed <ul style="list-style-type: none"> – h-entry – h-entry – h-entry
Entries with no h-feed	<pre><div class="h-entry"></div> <div class="h-entry"></div> <div class="h-entry"></div></pre>	<ul style="list-style-type: none"> • h-entry • h-entry • h-entry
Multiple feeds inside h-card	<pre><div class="h-card"> <div class="h-feed"> <div class="h-entry"></div> <div class="h-entry"></div> <div class="h-entry"></div> </div> <div class="h-feed"> <div class="h-entry"></div> <div class="h-entry"></div> <div class="h-entry"></div> </div> </div></pre>	<ul style="list-style-type: none"> • h-card <ul style="list-style-type: none"> – h-feed <ul style="list-style-type: none"> * h-entry * h-entry * h-entry – h-feed <ul style="list-style-type: none"> * h-entry * h-entry * h-entry

for which is preserved during parsing.¹⁹ This happened because the `e-content` is used within `h-entry` to identify the content of a blog post or article, and it turned out that most people creating events on their websites did so by adapting regular entries. One of my interview participants commented on how this influenced the specification:

There was this thing where people were posting events next to normal blog posts, which meant that they got a content property suddenly, sometimes. Because people just wrote it as a blog post, they had just switched out the entry tag for an event tag, but all the other stuff was the same. Then event suddenly had tags on it, or categories. Whatever other properties we usually find on a normal blog post were now also on the event, which isn't at all how an event was described by Microformats. But this is now how it's being used, so let's try and specify this as, okay this is the actual current usage of an event. We have to adapt.

(Interview participant #2)

As a result, the MF2 specification was revised to include a warning about `p-description`, “WARNING: Proposed to be replaced by e-content (re-used from h-entry),” as well as a link to a related discussion on GitHub (Microformats.org 2019). This revision occurred after I had written much of the first version of Yarns. As it turned out, I did a haphazard job of parsing events, since rather than following the spec, I wrote code for parsing events based on the structure I had observed on people's websites, which largely used the `e-content` property. As a result, I wrote a non-standard parser that worked with most actual implementations of events, but not with technically correct implementations that used `p-description`. This highlights that (1) I should have been following the spec, but also (2) the spec is subject to regular change, so should be reviewed on an ongoing basis.

These examples highlight how IndieWeb's principle of plurality shaped my work building Yarns. People publishing using IndieWeb building blocks can exercise considerable freedom in choosing how to structure and present Web content they publish, and this increases the complexity of writing a tool like Yarns that must make sense of messy data. Moreover, IndieWeb's “living standards” can change in response to observed behaviour, necessitating ongoing maintenance to keep software working. During an early debate in IndieWeb's chat, Çelik asserted that “optimizing for publishers over consumers has been a long-standing obvious design principle” because “there are far fewer programmers that write consumers rather than publishers.”²⁰ Autonomy for individual website builders results in additional labour for “consumers” such as Yarns, which aim to make sense of a plurality of IndieWeb sites.

4.5.4 Distributed collaboration and Microsub data structures

The second iteration of Yarns, which relied on the Microsub spec, was characterized by collaboration. At IndieWeb Summit 2018, I had a conversation with an attendee who contributes to a large number of IndieWeb WordPress plugins, one of which includes code for parsing websites. He suggested his Parser could be used in Yarns. To that end, he extracted the parsing code from his software, and released it as a library I could include in Yarns's code, called *Parse-This*.²¹ As described earlier, parsing was the most laborious part of building Yarns, so this was a good way to avoid us overlapping work on the same task.

¹⁹When parsing `e-` properties, the MF2 parsing specification dictates that the parser should return both an HTML version of the content, as well as a plain-text version (http://microformats.org/wiki/microformats2-parsing#parsing_an_e_property)

²⁰<https://chat.indieweb.org/2013-12-03#t138605524400000>

²¹see <https://github.com/dshanske/parse-this>.

Compared to the parser I had written, Parse-This can translate a more detailed set of MF2 properties for each post, such as a larger variety of response types (e.g. posts to indicate that the author *liked*, *read*, *bookmarked*, or *quoted* some content, as well as several other types of interactions). Additionally, Parse-This is more generalizable than Yarns, and thus is more likely to be useful to other developers in the future since it is more amenable to re-use and adaptation to future tasks. Finally, in addition to being more suited as a module or *building block* on the scale of the larger IndieWeb, Parse-This is internally modular to a greater degree than Yarns. Internal modularity of this sort is widely regarded as a feature of good software engineering (e.g. Hürsch and Lopes 1995). As I integrated Parse-This into Yarns, I started revising parts of my code to mimic its modular structure.

As a result of adopting Parse-This, my role in working on Yarns’s parsing functions became more devoted to testing and troubleshooting. When Parse-This was updated, I tested how it parsed a variety of sites and took detailed notes of errors. I attempted to resolve the errors and where I was successful, I proposed suggestions to be incorporated into Parse-This. Additionally, since the creator of Parse-This is the maintainer of several IndieWeb plugins for WordPress, he is knowledgeable about many technical considerations that were relevant for Yarns. Thus, our collaboration was not limited to working on Parse-This but extended to him providing guidance and suggestions about multiple aspects of Yarns’s design. I attempted to return the favour by testing some of his other IndieWeb projects while he built new features and acting as a sounding board to discuss a variety of design considerations.

As well as this direct collaboration, Yarns’s new Microsub design led to collaborative troubleshooting with developers of Microsub clients. Further, the connection between a Microsub server and client is supported by additional IndieWeb building blocks, such as *IndieAuth* so that users of a Microsub client can authenticate that they are allowed to access a particular Microsub server’s contents. Troubleshooting now required navigating multiple GitHub repositories, specification documents, and documentation, as well as communicating with a variety of other developers. From a research perspective, this was advantageous for developing insights about how IndieWeb building blocks fit together and the resulting influence on activities involved in building IndieWeb software.

In the first version of Yarns, troubleshooting parsing breakdowns highlighted the plurality of IndieWeb site architectures. Microsub introduced new breakdowns when communicating from Yarns to various clients, and this helped identify plurality among multiple tools built to the same standard. In theory, a standard provides a means for interoperability without additional coordination. Nonetheless, problems can arise when either client or server interprets the standard differently, or otherwise includes an error. I therefore tested Yarns against several Microsub clients to evaluate its functioning. In the case of Microsub, the earliest and most prominent implementations were written by the Microsub specification’s editor: A client called *Monocle* and a server called *Aperture*. I encountered circumstances where I thought Yarns was implemented correctly because it worked well with Monocle, only to realize that it caused problems with other clients that were stricter in terms of the formatting they could accept.

The most visible cases of breakdowns between Yarns and Microsub clients occurred when a client requested a list of feed items (i.e. posts) and Yarns responded in a format that the client did not support. Feed items are communicated from server to client using a format called JF2, which was edited by IndieWeb contributors and published by the W3C’s Social Web Working Group (Roberts and Marks 2018). The purpose of JF2 is to represent an MF2 feed using JavaScript Object Notation (JSON).²² JF2 is a clearly defined format, but allows some flexibility in data structures, which was a

²²JSON is a common data format for representing complex data objects as a string of text, which makes them easy to

major source of errors between Yarns and various Microsub clients. In fact, as will be discussed later in this section, the Microsub spec resolves some of the flexibility permitted in JF2, but both myself and other Microsub developers demonstrated confusion about this. I will first discuss how that confusion affected our development, and then address how and why we had misunderstood the Microsub spec and how this was resolved.

There are some cases in the JF2 spec where the format of a particular data property can vary according to its contents. Objects defined by the JF2 spec have properties, which can be of four types: “a single string, a single object, an array of strings, or an array of objects” (Roberts and Marks 2018).²³ The JF2 spec includes a statement that “Arrays that have only a single item SHOULD be condensed into only the single containing item” (Roberts and Marks 2018). Based on this statement, Yarns would format some properties as a string if the property had only one item, which caused an error among some clients that were expecting the property to be an array. This posed a challenge for clients because, to accept Yarns’s formatting, they would need to anticipate that some properties could be either a string or an array. In one case, an Microsub client for Android phones called *Indigenous* returned an error because it was expecting some properties to always contain an array, but Yarns returned them as a string. Upon identifying the source of the error, the programmer of *Indigenous* remarked:

So, the main question is: does the spec allow to send multiple variants of a property, e.g. a string vs an array? If so, then it’s a bug in *Indigenous*, if not, all these properties should return an array if it has a value.

Since this was a parsing issue, the programmer of *Parse-This* responded with a reference to the JF2 spec, according to which Yarns’s implementation was correct. The resolution was that this was understood as an error in *Indigenous*, and so *Indigenous*’ developer implemented a fix.

Throughout most of the development of Yarns Microsub Server, I interpreted that since Microsub was based on JF2, it embraced some of JF2’s flexibility. I was not alone in this interpretation, and in fact I developed this belief because during troubleshooting, others routinely referenced the JF2 specification as a means of resolving disagreements without being challenged. In mid-February 2019, in response to a chat discussion about how to structure some data, Microsub’s editor clarified that the Microsub spec should be interpreted more strictly than JF2:

the jf2 spec is possibly a bit out of date compared to what microsub is doing with it [...] in general we’re trying to reduce the number of different ways things happen with microsub

Shortly thereafter, two statements were added to the Microsub specification. The first asserted that Microsub is stricter than JF2, MF2, and other flexible data formats:

Where HTML or Microformats may provide multiple ways of representing content, Microsub instead has just one way of representing data. For example, in Microformats the “author” property may be a string URL or an **h-card**, but in Microsub it will always be a jf2 card.

The second of these statements indicated that examples presented in the Microsub specification were to be considered canonical, meaning that they represent the single correct way to structure Microsub data:

transmit and store.

²³In JSON a string is a sequence of characters, such as text. An object is an entity with multiple properties, such as we might describe a person using their name, age, height or other properties. An array is a list of items, which can themselves be either strings or objects.

(Note that the representations below should be treated as canonical examples, whereas the jf2 spec itself may provide multiple variations of values. For example, as of 2019-02-25, jf2 says that if there is one value of a property it should be returned as a string, but Microsub will always include the “photo” property as an array.)

This clarification offers a new view on the bug described earlier between Yarns and the Indigenous Microsub client. Indigenous expected Yarns to structure some data as an array, but Yarns used a string. We resolved this according to the JF2 spec, but the Microsub spec was later updated to clarify that Yarns’s implementation was incorrect, and so our resolution had diverged from the standard.

Had I encountered and reported errors between Yarns and the Monocle client I expect we would have arrived at a different outcome. Monocle was written by the editor of the Microsub specification and I suspect he would have clarified this misunderstanding about JF2 earlier. I perceived that the reason I had not observed such errors in Monocle was that, being written by the editor of the Microsub spec, it was simply more mature and consistent than other clients. Certainly, it was the case that Aperture is more consistent and reliable than Yarns. This belief had some merit, but with hindsight and further observation I have realized that some of the errors I encountered with other clients also occurred with Monocle, yet resulted in a more graceful failure. When Monocle received data in an invalid format, it simply omitted that data from the feed. The result was that the feed would be displayed with some information missing (e.g. the author’s name might be missing from a malformed entry). As a result, such errors were difficult to observe by eye because they are visually indistinct from cases where a post is simply missing some information. In brief, the same underlying error was encountered in both cases, but it was smoothed over in Monocle such that things could still operate invisibly as an infrastructure in the background. By contrast, the clients with which I had observed clear errors used *strongly typed* data structures, so when Yarns sent data in the wrong format, it would not load at all and an error message would be returned. The developer of a Microsub client called *Together* explained this to me during a conversation on IndieWeb’s chat: “that’s the thing with graphql, it is strict with its types so I can’t have the properties that are sometimes an array, sometimes a [string] and try to normalise stuff like that.”²⁴ This resulted in a harsher breakdown that drove us to investigate what went wrong.

There were a few other cases where Yarns encountered errors with a Microsub client. On these occasions I would typically post an issue to that client’s GitHub repository and resolve it through discussion with the client’s developer(s) and sometimes other IndieWeb contributors who interjected with advice or information. Both processes and outcomes of these discussions varied by circumstance. In some cases, one party demonstrated that the other had misinterpreted the spec, in which case person who had made an error would typically update their code to rectify their mistake. In other cases, the spec was not referenced at all, and resolution was more a process of dialogue and troubleshooting than deference to a canonical standard. In one such case, a developer asserted that the problem would be particularly laborious to address in the client. As a result, I modified my code to adapt to the client, even though I believed at the time that Yarns’s code was acceptable according to the spec. In other cases, conversations on chat could lead to impromptu ideas or bug fixes, for example in brainstorming sessions or in the process of troubleshooting tangentially related challenges.

Discussions of all sorts were crucial for articulating values. First, discussions were a way to present one’s views to other developers, encouraging the spread of particular design approaches. Furthermore, IndieWeb’s modularity nurtured working styles that supported values of collaboration. Collaboration

²⁴<https://chat.indieweb.org/dev/2019-06-21#t1561134117709400>

is implied in IndieWeb’s principles for building, but this is only codified as sharing documentation and source code. The prominence of IndieWeb’s events and chat rooms demonstrate that discussion and collaboration are central features of this community, and in fact so central that they are taken for granted. One indicator of this is that when I asked interview participants about how many hours per week they spent on IndieWeb-related activities, it was common for them to separate time spent in IndieWeb’s chat from actually working on software, websites, and other projects. Thus, it is worth noting that although IndieWeb’s principles for building emphasize individualism, in practice, IndieWeb building is highly social and collaborative.

A final dimension of collaboration is articulated through ties of obligation to others. Once I had the basic foundation of Yarns Microsub Server in place, my development efforts were largely spent accommodating others. Most obviously, on the occasions that someone who was using Yarns reported a bug, fixing that bug became my top development priority. And as I worked closely with the developer of *Parse-This*, much of my time was spent adjusting Yarns’s code to take advantage of parsing features he had developed. And similarly, I could see that much of his development time was focused on helping me (for which I hope I’ve suitably expressed my appreciation). As I write this, the most urgent outstanding issue on Yarns’s GitHub is entirely to support someone else. Much to my embarrassment, it’s been sitting in my to do list for weeks, as I focus my time on writing this dissertation rather than doing any development. My dominant emotion toward that task is guilt at my lack of timeliness in addressing that obligation. This is not an emotion I associate with the self-direction implied by IndieWeb’s stated principles, but one that derives from social bonds formed through working closely with other IndieWeb contributors.

4.6 Discussion and conclusion

I will conclude this chapter with two brief discussions. First, I will discuss challenges for maintaining reflexivity while developing Yarns development and explain how these affected Yarns’s relationship with values. Second, I will return to the research questions posed at the beginning of this chapter to address what values are evident in IndieWeb’s building processes and how they are employed.

4.6.1 Reflection and balancing individual with communal values

As soon I began developing Yarns, I experienced a familiar pressure to solve instrumental challenges. An early excerpt from my field notes reads:

I’m feeling a bit lost working on this stuff [developing for WordPress] since it’s been awhile, but am also finding that I can slip back into the right frame of mind with some work. This requires entering a logical state of mind where I follow variables/functions etc. through the code). This type of thinking does tend to close off “critical” reflection since it requires an intense focus on just getting the thing to work. (September 18, 2017)

The “right frame of mind,” as I put it, required me to focus on solving problems methodically and with “intense focus.’ However, those periods of focus threatened my capacity to take a broader view of how specific technical challenges and procedures related to the values I was trying to achieve. I attempted to address this in my field notes by repeatedly recording my value-oriented goals and connecting them

with design decisions. Nonetheless, upon reviewing my notes I can observe that there were periods of development in which I was focused on instrumental aspects of development.

For example, I regularly asserted in my field notes that the main value I wanted to achieve with Yarns was *ease-of-use* for people with varying levels of technical experience. This commitment was one of the rationales for building Yarns as a WordPress plugin; Most WordPress users are familiar with how to install and use plugins, so Yarns would not require any new training or knowledge to set up. However, when I first started using IndieWeb plugins for my WordPress site, I had been confused by how various building blocks fit together. I had to install several plugins for various IndieWeb features and use an IndieWeb-compatible theme, which meant changing the visual style of my site. I did not have a strong understanding about how building blocks were related, and this made it difficult to fix bugs that arose during interoperability. IndieWeb’s ecosystem of building blocks, including the combination of WordPress plugins I used on my site, has matured significantly since my early encounters. As a result, newcomers’ early experiences with IndieWeb are smoother than they used to be. Nonetheless, my early frustrations shaped how I approached ease of use as a value while designing Yarns.

Based on that frustration, I wanted Yarns to be able to operate independently of other IndieWeb structures. I felt this would be easier to understand for newcomers to the IndieWeb. I designed the first iteration of Yarns, *Yarns Indie Reader*, to function without other IndieWeb features installed, and then seamlessly integrate relevant features if available. For example, Yarns Indie Reader has options to respond to one’s newsfeed with *replies* or *likes*. If a user had the *IndieWeb Post Kinds* plugin installed, then Yarns Indie Reader would hook into that plugin to define those posts’ *kind* appropriately, which resulted in those posts being marked up with appropriate MF2. As described earlier in this chapter, that is what allows IndieWeb sites to parse incoming Webmentions (e.g. to display *Bob liked this post* or *Bob replied to this post*), and is also useful for software such as feed readers to identify and filter posts by type (e.g. to display a feed of only photos, or only replies). If IndieWeb Post Kinds is not installed Yarns will fall back to use a built-in feature of WordPress, resulting in a steep compromise in terms of integration with IndieWeb.²⁵

Once I began the second iteration of Yarns, *Yarns Microsub Server*, this effort to provide a self-contained experience was abandoned altogether. Because the Microsub specification divides client and server, Yarns must now be paired with a client. Users must now make a decision about which client to use and therefore must have a general understanding of the client-server model. Logging into a Microsub client requires granting the client permission to access the servers’ contents, which uses IndieWeb’s *IndieAuth* protocol. Replies, likes, and other responses to one’s newsfeed must be communicated from the client to a server, which is done using the *Micropub* specification. Thus, whereas the first version of Yarns could stand on its own, Yarns Microsub Server requires users to install additional IndieWeb plugins (*IndieAuth* and *Micropub*), then log in to a third-party client of their choice. Once all these steps are performed, the user experience provided by Yarns as a Microsub server accompanied by a variety of clients is arguably much better than what I had built originally. However, the requirement to negotiate multiple building blocks upon setup directly conflicted with my initial plans.

This compromise came as a surprise. My prioritization of ease-of-use had slipped away without me realizing. As I reviewed my field notes and GitHub commits from Yarns’s development it became evident that they became sparser and more focused on technical troubleshooting after I began using

²⁵Yarns Indie Reader falls back to WordPress’s *Post Formats* feature if the *IndieWeb Post Kinds* is not installed. By default, post formats do not include *replies*, *likes*, or several other types of posts commonly used to interact among IndieWeb sites.

Microsub. To some extent, this is a personal failing in that I became swept up in the technical aspects of building Yarns. This reflects a tendency trained through my earlier experiences working as a freelance Web developer, where working to deadline was usually a higher priority than critical reflexivity. However, I will argue here that another factor was a division of labour articulated through widely held software development conventions as well as IndieWeb’s modularity.

One of the effects of working closely with the developer of Parse-This, was that I developed an appreciation for the clear, modular structure of his code. Based on this appreciation, I attempted to make Yarns’s code more modular in the same way. Here I was roughly following a foundational software engineering convention called *separation of concerns*, in which code is structured into sections that each deal with one *concern*—i.e. each section deals with a single aspect of the program (Hürsch and Lopes 1995). This is an entrenched concept in software engineering and is used almost everywhere. It offers significant benefits for re-using sections of code, avoiding duplication, and overall writing code that is easier to maintain. From an instrumental perspective, my code increased in quality as I adopted this approach, yet this was a challenge for ongoing reflection. After an initial period of conceptualizing how the overall program should be structured, I found myself working on individual functions in isolation from one another. This in turn drew my attention away from how multiple *concerns* might relate to one another, excepting those that were computationally linked, for example by passing a variable from one function to another.

This approach operates at multiple scales. On a micro level, separations of concern structured sections of code within a project. During the process of working on Yarns, this made certain features easy to take for granted, which in turn encouraged me toward a task-oriented style of work through which I lost site of the *ease-of-use* value as I had defined it earlier. On a meso level, IndieWeb’s building block approach is an articulation of the same *separations of concerns* concept. This demonstrates divisions of labour that limited my access to decision-making at two levels. First, the decision to encourage modularity was well-established in advance of my work on Yarns, both at the level of IndieWeb building blocks and as a general software convention. Second, the boundaries between IndieWeb building blocks are defined by standards; Once I set out to build a Microsub server, I was not engaged in decisions related to Microsub clients (except through bug reporting, feature requests, and other troubleshooting). Even had I decided to build a Microsub client in addition to Yarns, the conceptional cut between client and server would remain a decision that was out of my control.

Given IndieWeb’s lack of a managerial structure, these divisions of labour were neither forceful nor permanent. There is nothing that would have prevented me from ignoring Microsub and continuing with *Yarns Indie Reader* as an all-in-one solution. However, interactions with other IndieWeb contributors and documents such as IndieWeb’s *principles* established norms about IndieWeb’s approach, which I embraced. Having done so, I found that at the moment I was working on a particular component’s code, the rest of the system was presented as an infrastructure operating invisibly in the background (Star 1999). Thus, where it was still possible to access other parts of the system, modular architecture constituted a technical tie (Damarin 2013) that shaped my work practice. The decision to take up this architecture was influenced by human ties through encouragement in that direction, and postsocial ties because I imagined potential Yarns users as expecting it to support Microsub. This illustrates the extent to which technical, human, and postsocial ties co-constitute one another, since the existence of the Microsub standard articulated a norm about anticipated users, which contributed to its communal uptake.

4.6.2 Employed values

Here I will return to the research questions asked at the beginning of this chapter:

- **RQ1:** *What values are central to the IndieWeb, and how are they defined, selected, and critiqued?*
- **RQ2:** *How are values employed in design activities?*

These questions are tightly linked, such that the way values are employed in design activities contributes to their definition and selection. Thus, I will address them together.

I began this chapter by identifying values related to intended *outcomes* of IndieWeb software. Autonomy and self-empowerment are presented as overarching goals, defined and selected from community members' long held perspectives of the Web, and further articulated in principles for building. Outcome-oriented principles—*#1. Own your data; #9. Longevity; #10. Have fun*—specify that these values are defined by individuals controlling what data they put online. The overall strategy for this is to use one's own website instead of relying on a third-party platform. Clearly, this is a fundamental design decision that defines IndieWeb's overall approach.

These values are also articulated through principles about *processes* for building, with autonomy being particularly salient. Specifically, individuals are encouraged to build for their own needs (*Principles #3. Build what you need; #4. Use what you make*), and to start building right away rather than waiting for consensus. This results in a system that is modular and pluralistic (*Principles #8 and #10, respectively*), with interoperability to be figured out after individuals have been free to develop a variety of approaches. Here, autonomy is employed as a value for defining processes of building. Further, I've described how a value of benevolence is employed in principles related to documenting one's work and sharing source code (*Principles #5 and #6*).

These process-oriented and outcome-oriented principles are closely related, and in fact co-constitute one another. The clearest way to illustrate this is to point to the absence of IndieWeb projects related to privacy. Privacy has long been a central concept in research about values and design in information technologies (e.g. Tang 1997; Friedman, Howe, and Felten 2002) and is a clear priority in the design of other decentralized Web projects such as Mastodon, Solid, and Diaspora*. By contrast, IndieWeb sites are on the public Web and there is no widely deployed mechanism for sending private messages in spite of interest to do so. IndieWeb's wiki identifies that there were at least 10 sessions about privacy at IndieWebCamps between 2014-2019 (IndieWeb.org 2020j), and proposals for specifications that would facilitate private messages have been implemented by a small number of IndieWeb participants but never widely deployed.²⁶

One of the reasons private posts have been such a challenge is that they require a level of coordination that is difficult to achieve with an individual and iterative development style. Making sure private messages are secure requires tight coordination among the website from which a private message is sent, the website of the person authorized to view it, and potentially a third-party software in which they will access the message (such as a Microsub server).

You need a site that can provide private posts with that protocol, which is one or two pieces of software. [...] Then you have the site of the person [that accesses that [...]] and then you need the software that actually does the accessing. [...] You need all of them to get to something useful. (Interview participant #1)

²⁶E.g. Private Webmention (IndieWeb.org 2019h) and AutoAuth (IndieWeb.org 2020a)

In brief, designs devoted to privacy are unlikely to emerge from a development style in which individuals build whatever solution works for them and then figure out consensus after the fact. Rather, privacy and its requisite security demand consensus up front.

Building a feed reader demonstrated a similar set of challenges. A feed reader is only useful when there is a consensus about how sites should be structured so as to be parsable (e.g. a shared agreement to mark up IndieWeb sites with MF2). Microsub intensified the requirement for consensus because it is only useful when a client and server agree about how information is exchanged between them. Achieving this consensus requires compromises to plurality as an expression of developers’ autonomy. This is reflected in the fact that the Microsub standard was written prior to any implementations, rather than capturing and codifying existing practices. Further, working through bugs between clients and servers required coordination and agreement among developers. This extended beyond the *benevolence* value articulated by documenting and sharing one’s work. This coordination demonstrated a value of *conformity values*, which Schwartz describes thusly:

Conformity values derive from the requirement that individuals inhibit inclinations that might disrupt and undermine smooth interaction and group functioning. As I define them, conformity values emphasize self-restraint in everyday interaction, usually with close others. (S. Schwartz 2012, p. 5).

Schwartz’ theory of basic human values situates conformity values as oppositional to self-direction values such as autonomy (S. Schwartz 2012). However, in this instance it is evident that conformity in one situation can support autonomy elsewhere. In this case, IndieWeb’s overarching commitment to autonomy was supported in two ways. First, Microsub readers aim to facilitate individuals owning and controlling their feed reader experience. Second, the client/server division of this spec increases the feasibility for individuals to build their own reader software, or at least part of it. These articulations of autonomy were pursued by constraining individual autonomy during development. This speaks to the value of approaching values as hypotheses to be employed during design processes (JafariNaimi, Nathan, and Hargraves 2015). For situations such as building an IndieWeb reader, autonomy has not served as a sufficient hypothesis. This is illustrated by a long history of efforts to build IndieWeb feed readers with autonomy as a driving value. Several IndieWeb contributors were developing reader software as early as 2012²⁷. Although many of these individual projects resulted in functional software that was at least useful to their creators, none achieved widespread use and readers remained a gap in IndieWeb’s collection of tools until Microsub heralded “the year of the IndieWeb reader” in 2018 (Çelik 2018). Conformity is a property of all standards, but notable here because Microsub conflicts with IndieWeb’s convention of generating standards from observed behaviour.

In sum, IndieWeb’s premise highlights individual autonomy as a core value. On an individual scale, the model of autonomy is simple—using a personal website that one owns and controls contributes to autonomy over one’s online identity. This is accomplished by customizing one’s website structure, style, content and other features. On the other hand, coordination across IndieWeb sites requires some degree of conformity. They need to speak the same language or rely on an agreed upon means of translation if they are to talk to one another. To accomplish this, IndieWeb’s approach generally prefers simple, modular standards based on observed practice, only resorting to top-down standards where previous efforts have been unsuccessful. As a principle (specifically, principle #7), standards are to be “the absolutely simplest,

²⁷See the revision history for https://indieweb.org/feed_reader at https://indieweb.org/wiki/index.php?title=feed_reader&action=history

easiest, and most minimal protocols & formats sufficient to support [a desired user experience], and nothing more” (IndieWeb.org 2020i). In essence, the balance between individual autonomy and larger-scale interoperability is an example of Franklin’s (2004) distinction between holistic and prescriptive technologies. Individual autonomy is achieved through holistic approaches to building where individuals control every aspect of their labour process. However, prescriptive technological processes support the ability to make structures of greater complexity and coordination than can be achieved through purely holistic approaches. IndieWeb’s balance between holistic and prescriptive technologies concerns not only the scope of what can be accomplished, but also the scope of who can participate in that accomplishment. Holistic technologies both enable and demand that one has control over the labour process. Thus, they only afford autonomy to those with the requisite skillset. This raises a question about how IndieWeb’s material design may contribute to inclusion or exclusion of certain people, to which which I will return in Chapter 6.

4.6.3 Conclusion

This chapter has presented results about individuals’ building activities for the IndieWeb. I have identified how overarching commitments to autonomy and self-empowerment are articulated in IndieWeb’s principles for building, and discussed how technical standards and IndieWeb’s *building block* approach act as technical ties through which individuals’ activities are structured to contribute to IndieWeb’s broader infrastructure. The latter section of the chapter has recounted a personal narrative of building for the IndieWeb, highlighting how interoperability with other IndieWeb projects, standards, and collaborating with community members shaped my processes of building.

In sum, through most IndieWeb development, commitments to individual autonomy and self-empowerment are evident as both outcomes and as guiding principles for building processes. However, it is also evident that individual and iterative building styles limit the range of possibilities for other values that can be expressed through building. Specifically, when autonomy values are employed to define building processes, they pose challenges for achieving outcomes that rely on community consensus. By reflecting on my own processes of building IndieWeb software, I have identified limits of individual autonomy and plurality, which has highlighted that even in individually-focused acts of building IndieWeb software, communal values and structures are profoundly important.

This leaves open questions about how power and influence may operate among IndieWeb’s community. Those questions will be addressed in Chapter 5, which presents a structural overview of the community through its logged discussions and GitHub activity, and in Chapter 6, which investigates factors contributing to influence and exclusion.

Chapter 5

Mapping IndieWeb’s community structure with logged data analysis

This chapter describes the structure of IndieWeb’s online community by reporting results from analyzing archives of logged data. The data sources for this analysis are IndieWeb’s publicly chat archives and collected logs of GitHub activity related to the IndieWeb. The procedures for collecting and analyzing this data were described in Section 3.3.

Chapter 4 established that individuals’ design decisions are influenced by technical artifacts and shared beliefs established elsewhere in the community.

Here, I expand upon that analysis to describe the community structure in which IndieWeb’s participants build their personal projects, contribute to standards, and discuss IndieWeb’s future directions. By doing so, I address the first part of **RQ3**: *Is power over planning and executing design activities centralized or decentralized among stakeholders, and is anyone included or excluded as a result?*

Specifically, in this chapter I investigate whether there is centralization or decentralization in IndieWeb’s chat discussions and development activity in IndieWeb-related GitHub repositories. The results presented here are organized as follows. First, I present summary statistics about IndieWeb’s chat logs, focusing on users’ frequency and duration of chat activity. Second, I identify topics of discussion using Latent Dirichlet Allocation (Blei et al. 2003). I then describe the structure of IndieWeb’s network of GitHub repositories. Finally, I combine the preceding three analyses to identify types of chat activity that are predictive of contributing on GitHub. Across these results, I present an overview of IndieWeb’s community structure. I identify that chat activity is generally concentrated among a centre of extremely active participants, and that members of that centre are more likely also be active on GitHub. These active users form a soft centre of influence over IndieWeb’s overall direction, including what is discussed in its online spaces and how. Nonetheless there is evidence that projects on GitHub are generally carried out by individuals, indicating that processes of building IndieWeb software are largely decentralized across an assortment of personal projects. Ultimately, this chapter works in tandem with Chapter 6 in addressing RQ3. By identifying points of centralization and decentralization, this chapter provides a foundation for Chapter 6 to analyze factors contributing to inclusion and exclusion.

5.1 Summary statistics about IndieWeb’s Chat

Table 5.1: Overview of chat log data

Earliest message	2011-02-12 18:50:00
Latest message	2019-07-24 19:56:45
Number of messages	923,634
Number of users	2,897

This section presents summary statistics about IndieWeb’s chat archives. Table 5.1 presents the start and end date of the chat logs used in this analysis, as well as the total number of messages and unique users present in this data. Note that, as described in Section 3.3.1, usernames were processed to remove common duplicates, so variations of the same name in formats such as *‘jackjamieson’*, *‘[jackjamieson]’*, and *‘jackjamieson[mobile]’* were merged into a single user.

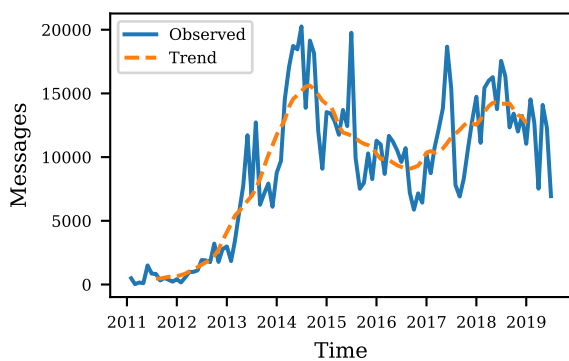


Figure 5.1: Number of messages per month

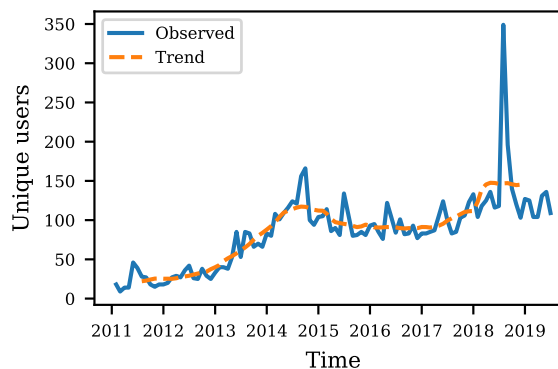


Figure 5.2: Number of unique users per month

Figures 5.1 and 5.2 illustrate the number of messages and unique users per month, respectively. Both figures show a general pattern of growth, with a peak in mid-2014, a small decline from 2015-2016, and a second period of growth from 2017 through 2019 when the data was collected. Additionally, Figure 5.2 shows a dramatic spike in the number of users during August 2018. This is an anomaly resulting from a spam attack that affected Freenode, which hosts IndieWeb’s IRC channels. After the initial growth period, IndieWeb’s chat channels are posted to by approximately 100-150 uniquely identified people each month.

There are a number of factors that possibly contribute to the increase in growth from 2013 to 2015. During this period IndieWeb received media coverage from a variety of sources, such as Wired Magazine (Finley 2013; Sterling 2013); and New Statesman (Crawley 2013) in 2013, Slate (Gillmor 2014b), Gigaom (Ingram 2014a; Ingram 2014b), and The Atlantic (Gillmor 2014a) in 2014. This period also saw the launch of some significant IndieWeb services. For example, *Bridgy*, a free service for syndicating from a personal website to popular social media platforms, then “backfeeding” comments and other responses launched an IndieWeb-compatible version in 2013. This significantly eased the technical challenge of

adopting IndieWeb’s *POSSE*¹ approach. Another notable release during this period was *Known*, an open publishing and community platform project that adheres to IndieWeb principles. Known was funded by Matter ventures and was launched on September 11, 2014 (IndieWeb.org 2020d). Finally, the number of IndieWeb events increased significantly during this period. The first Homebrew Website Club was held in San Francisco on November 20, 2013, and quickly became a (usually) biweekly series of events held in many different cities. Further, the number of cities that hosted IndieWebCamp events increased during this period. IndieWebCamps were held in 2 cities in 2012, 3 cities in 2013, 6 cities in 2014 and 2015, and 8 cities in 2016. Appendix G shows a figure of IndieWebCamp locations over time.

Table 5.2: Count of messages per user posted to IndieWeb Chat

	Mean	Median	SD	Min	Max
Total messages per user	318.82	5	4295.33	1	169,780
Monthly messages per user	106.64	10	312.92	1	3927

Table 5.2 shows the distribution of messages per user, both overall and monthly. This table shows that a small number of especially active users contribute the majority of posts. The median total post count is 5, and the median monthly post count is 10. The latter metric is calculated by averaging the number of messages per each observation of a single month’s activity per user, which means that a user who posts to IndieWeb’s chat during multiple months is counted for each month. That the monthly post count has a higher median than the overall count indicates that people who participate over a greater number of months are generally more active than those who post only during one month.

5.1.1 Length of chat participation

Table 5.3: Number of active months per chat user

Active months	Percentage	N
1	71.35	2067
2	10.25	297
3-6	10.18	295
7-12	3.80	110
>12	4.42	128

The number of active months for each chat user was calculated, where an active month was one in which the user posted at least one message to IndieWeb’s chat. The median number of active months was 1, with a mean of 2.99, and a maximum of 101. Table 5.3 indicates that 71.35% of chat users were only active for a single month, 10.25% for 2 months, 10.18% for 3-6 months, 3.80% for 7-12 months, and 4.42% of chat users posted to IndieWeb’s chat during more than 12 distinct months.

¹Publish on Own Site, Syndicate Elsewhere. see <https://indieweb.org/POSSE>

5.2 Results: Topic Modelling

Having identified some basic statistics about the structure of IndieWeb’s chat community, this section presents results of a topic model analysis of messages posted to chat. Based on the procedure described in Section 3.3.2, a topic model was generated, which classified chat conversations into 16 topics.

The top-30 most relevant terms for each topic, as classified by *pyLDAVis* with a λ value of 0.6, are presented in Appendix H. While these terms provide a beginning for understanding the contents of each topic, several other metrics were used to interpret the results of the topic model. These include: (1) the distribution of topic weight scores across documents classified with each topic, (2) the distribution of meaningful key terms across each topic, and (3) qualitative readings of representative and randomly selected documents.

5.2.1 Topic probability distribution across documents

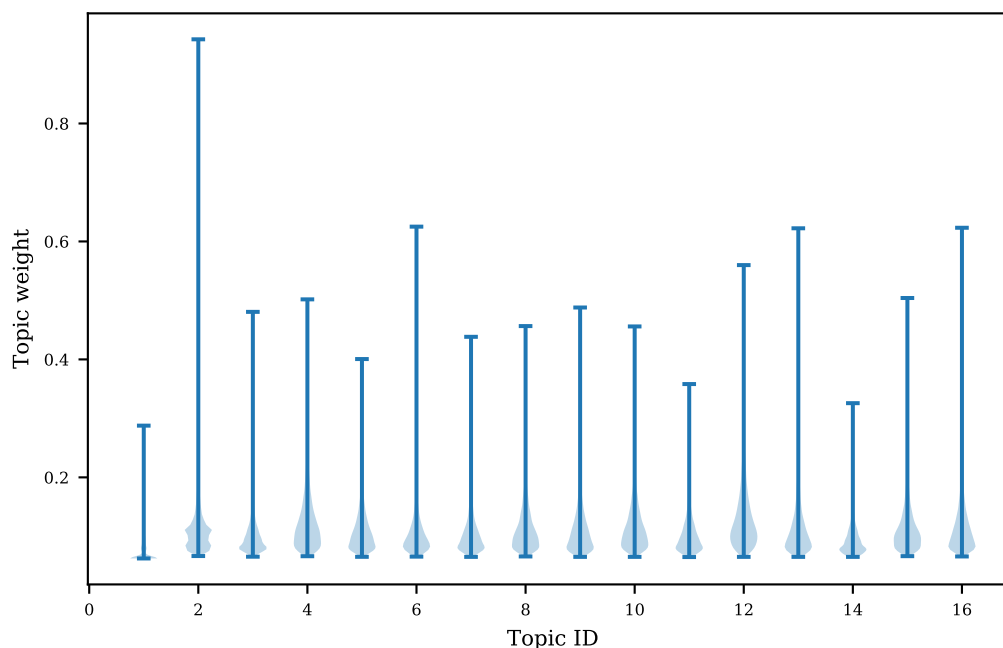


Figure 5.3: Distribution of dominant topic weights per document.

For each document, the LDA model calculates a topic probability distribution. The topic with the highest probability in a given document was identified as that document’s dominant topic. Figure 5.3 presents a violin chart showing the distribution of topic weights per document for each dominant topic. The most important finding in this figure is that Topic 1 has very a low topic weight per document. The median topic weight for documents classified with topic 1 as dominant is 0.0625. Since this is $\frac{1}{16}$ and there are 16 topics in the model, when a document was classified with topic 1 it is almost always the case that there was no actually dominant topic.

5.2.2 Distribution of keywords across topics

Several lists of key terms were generated to identify pre-defined topics of interest: Principles, IndieWeb standards, names of IndieWeb projects, and events. These lists are presented in Appendix E. After categorizing documents by their dominant topic, the distribution of terms from each list across topics was calculated.

Principles related keywords

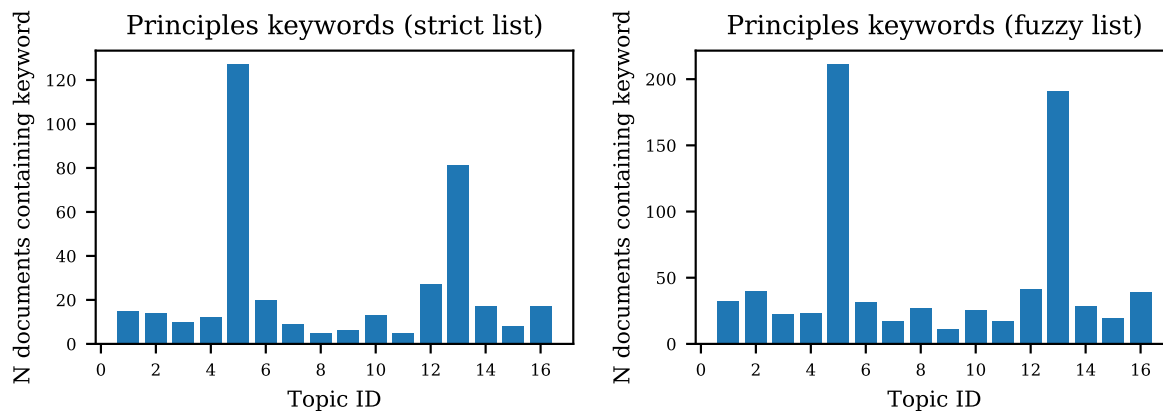


Figure 5.4: Distribution of principles-related keywords across topics. *Left:* Strict list of principles-keywords. *Right:* Fuzzy list of principles keywords.

Figure 5.4 presents the distribution of terms related to IndieWeb’s principles page (<https://indieweb.org/principles>) across the topic model. The left chart displays results based on a strict list of terms (only “principle” and “principles”) and the right chart uses a more expansive list of terms as defined in Appendix E. These charts show that terms related to IndieWeb’s principles were concentrated in topics 5 and 13. Additionally, these figures demonstrate that explicit use of those terms is relatively uncommon in IndieWeb’s chat, demonstrated by the small scale of the Y-axis for principles-related keywords when compared to charts for other keywords sets later in this section.

Standards related keywords

Figure 5.5 presents the distribution of terms related to IndieWeb’s technical standards across the topic model. This shows that terms related to IndieWeb’s standards were concentrated in topic 4, and also had a strong presence across topics 8, 9, 10, and 12.

IndieWeb project related keywords

Figure 5.6 presents the distribution of terms related to various IndieWeb projects across the topic model. As defined in Appendix E, these keywords consist of the names of various individuals’ IndieWeb software projects. These terms were most prominently topics 4, 9 and 16, though were distributed relatively highly across other topics as well. Additionally, there is significant similarity between Figure 5.5 and Figure 5.6, showing that the topic model has not disentangled conversations about standards from conversations about specific projects. In practice, keywords about both

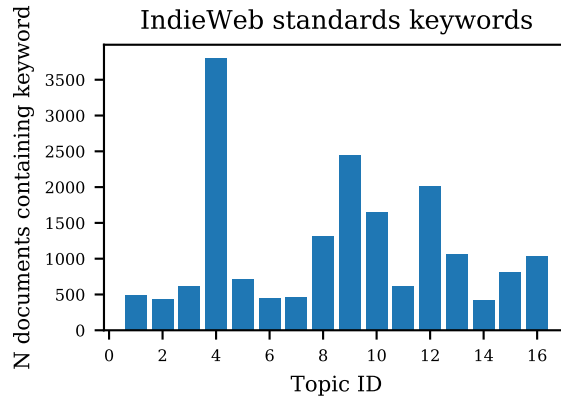


Figure 5.5: Distribution of standards-related keywords across topics.

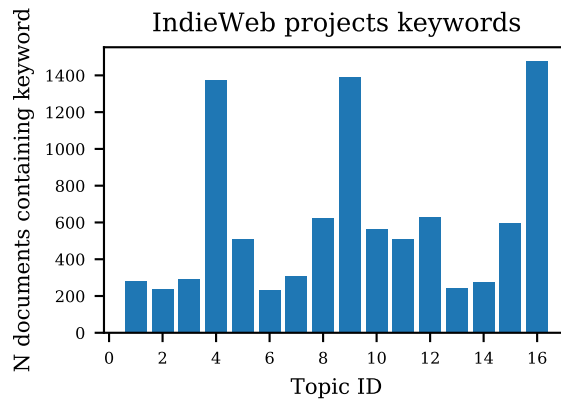


Figure 5.6: Distribution of IndieWeb project related keywords across topics.

Event related keywords

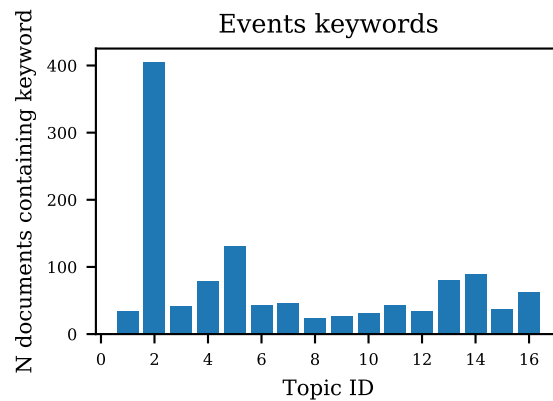


Figure 5.7: Distribution of event related keywords across topics.

Figure 5.7 presents the distribution of terms related to IndieWeb's technical standards across the

topic model. This shows that terms related to IndieWeb’s standards were strongly concentrated in topic 2.

5.2.3 Labelling and grouping topics

Topics were labelled based on the contents of their most representative documents, as well as a random selection. Some topics were linguistically distinct but overlapped in meaning. For example, multiple topics concerned different IndieWeb technologies. Although these use different terminology and so were categorized separately in the LDA model, for the purposes of this analysis it was more meaningful to group them together into an ‘IndieWeb Building Blocks’ topic group.

This section describes how topics were labelled and grouped into the following categories: “IndieWeb building blocks,” “Web dev (general),” “Defining IndieWeb,” “Events,” “Online community management,” and “Other topics.” For categories that include multiple grouped topics, an explanation of each topic is presented and accompanied with one or more excerpts of representative documents. Each “document” consists of 30 minutes of one user’s chat messages, with no punctuation indicating breaks between individual messages. Where documents include identifying references such as usernames, these references have been redacted with “[name].”

“IndieWeb building blocks”

Six topics were grouped into a category called “IndieWeb building blocks.” This topic group focused on discussions related to IndieWeb-specific development, and particularly to IndieWeb standards and projects. Conversations classified here are characterized by the use of highly specific terminology, much of which originates within the IndieWeb, such as names of standards and applications. As a result, it is generally possible to delineate topics within this group from one another, though for the purposes of this dissertation’s analysis it is more meaningful to group them. The following topics were included in this group: 4, 8, 9, 12, 15, and 16.

Topic 4: Micropub and IndieAuth. Documents labelled with this topic were clearly identified as relating to Micropub and IndieAuth. Micropub is a specification for publishing to one’s website from a third-party application. IndieAuth is a protocol used to validate one’s authorization to post in this manner.

E.g. [...] the micropub endpoint that would validate the bearer token would be the endpoint on the users site so it should only have one token endpoint In other words. If I log in to a micropub client it requests a token from my token endpoint. [...]

Topic 8: WordPress. Documents labelled with this topic was clearly associated with IndieWeb related development for WordPress. WordPress is not itself an IndieWeb technology, and some conversations in this topic refer to general WordPress issues more than IndieWeb specific concepts. However, this has been classified as part of the *IndieWeb building blocks* group because the majority of conversations are about IndieWeb-specific plugins and themes.

E.g. ok, that is a good point! I removed it from the IndieWeb plugin and perhaps we should release a new version without the uf2 plugin and re-add either the mf2-feed or wp-uf2 plugins or both if they really work to keep confusion low

E.g. *Bear in mind, the developer of Webmentions and Semantic Linkbacks also did the Indieweb plugin [...] I did Indieweb Custom Taxonomy, and its successor Indieweb Post Kinds, as well as some other Indieweb plugins, Semantic Comments, Syndication Links. [...] The WordPress Syndication Plugin was [name]. [name]: If they were all together, we couldn't develop them independently*

Topic 9: Webmentions. Conversations classified as this topic concern Webmentions, one of IndieWeb's most prominent building blocks.

E.g. *!tell [name] had another thought re: dates from webmentions: receivers of webmentions should keep track of when a webmention for a URL was *first* received (CREATE) and *most recently* received (UPDATE), and may use that instead of (or with) the explicit dt-published / dt-updated respectively. [...]*

E.g. *here's an example of a receiver test for webmention.rocks where i'm not sure the best way to test for this: "Verifies that target is a valid resource for which the receiver accepts Webmentions" how do I know what resources the receiver accepts webmentions for? I could assume that no legitimate receiver would accept webmentions for example.com but that isn't necessarily true [...]*

Topic 12: Microformats. This topic concerns Microformats, the primary method for marking up machine-readable content among IndieWeb's community.

E.g. *[name] - they work just like nested HTML elements do you have a specific question? [name] - that's a *very* confusing statement. please restate it in a positive way [name] you can put microformats property class names on any descendant inside a microformats root class name element, WITHOUT any intervening microformats root class names in the descendent chain [...]*

On <http://indiewebcamp.com/authorship#Determining>, under the rel-author sub-section, I'm not sure I understand the difference between #2 and #4. 2: "if there's an h-card with a u-url == u-uid == that page's URL, use that h-card for the author." 4: "otherwise if there's an h-card on the post's page with a u-url == rel-author link's href, use that h-card for the author." "u-url == u-uid == that page's URL" seems the same as "u-url == rel-author link's href", just more verbose. [...]

Topic 15: Post types. This topic consists of conversations about handling various post types on IndieWeb sites. This topic has been classified as part of the "IndieWeb building blocks" group because these conversations focus on managing post types in a way that can be parsed and understood by IndieWeb tools.

E.g. *The way I understand them, "Photo" vs "Note with Photo" are pretty distinctly different. A "Photo" post is basically a photo without any other content (i.e. photo filename = e-content, p-summary, p-name, u-photo) or where the photo is the entire e-content (and u-photo) accompanied by a p-summary / p-name. A "Note with Photo" is a note post that includes a photo referenced in the e-content that is attached and marked up as u-photo.*

Topic 16: POSSE. This topic refers to POSSE, IndieWeb’s practice of Publishing on one’s Own Site, then Syndicating Elsewhere. A large portion of these conversations relate to Bridgy, a popular tool for POSSE among IndieWeb’s community.

E.g. [...] reading /quotation#reply has me convinced that i want to post an in-reply-to on my own site but when i POSSE to twitter i want it to appear as a /quote_tweet, not a /reply i want it to be public on my timeline i’ve got an example. [...] a /reader subscribed to my site would show that context and my text, and that would be what i want. however, when i POSSE this through bridgy, it becomes a reply to the original tweet, which would not show up in my main list of tweets. [...]

[...] But bridgy is detecting that my tweet is in reply to my own previous tweet even though they have another persons’ tweet in the middle [...] so chained conversations would sort of not work optimally I am not chaining conversations at all on my side... so right now, there would be my post and [name]’s direct reply below it my next reply to [name]’s post shouldn’t be a reply to my original but maybe twitter does not handle reply to replies and reply to original separately[...]

“Web dev (general)”

Three topics were categorized as being related to general Web development. It was difficult to meaningfully distinguish topics within this group except to identify that they concern various activities and concepts related to building websites or Web applications. At times, conversations in this group refer to more specific IndieWeb development. And given that these conversations occur on IndieWeb’s chat, they are related to IndieWeb building blocks in a general way. However, they have been classified separately from “IndieWeb building blocks” because a significant portion of the documents reviewed are related to technologies developed outside of IndieWeb’s community, such as servers, content management systems, and Web frameworks and languages. The following topics were included in this group: 7, 10, 11.

Topic 7: General Web development 1. Documents with this topic concern a variety of Web development activities and concepts.

E.g. So I added a dark mode to my website for macOS Mojave users with the Safari Tech Preview (and eventually after the new Safari version, any Mojave user with dark mode). The biggest question is, should I provide an actual “dark/light” mode switcher to allow someone to switch between themes?

E.g. Working with Laravel at work now, so investigating a switch with [name], but not yet sure about it all. I do want to fix the general buggyness of my site (don’t we all) haha Yeah I notice I don’t use any UI from the Kirby CMS now, so in a way I’m not really using the Kirby CMS, only the Kirby Framework. And it’s just not that good at being a framework.

Topic 10: General Web development 2. Like topic 7, this topic concerns general Web development activities.

E.g. [...]if we are using DOM spec’s textContent (as I assumed, and as I write in the issue) that is fine but should be called out. And if that is decided, a bug should be filed on the PHP

parser which isn’t doing so at present. Again because of reasons captured in the issue. HTML spec builds on DOM spec, but yes. So the PHP parser is wrong per-spec, but the PHP parser does what at least 2 users (the one who opened the issue, and [name]) want.

E.g. Yeah, and associative arrays can also use name: value, name: value, as JSON I always found YAML to be a bit of a mess because of the huge variations in how you can write things. Have also had trouble finding proper libraries to handle it in PHP, most only implement an older version or a subset of the YAML spec :/ Reading the Wikipedia article, any valid JSON document should also be a valid YAML 1.2 document So in a way, mf2 parsing already creates YAML ;)

Topic 11: Deployment and Web servers. These conversations generally refer to activities and technologies related to deploying websites to a server. These are characterized by references to git and GitHub, Jekyll—a “static site generator” commonly used when hosting websites on GitHub Pages—and related technologies.

E.g. really? huh. did you create a new branch since the first PR? or were you squashing or something locally? try git push -f to rewrite history with your local commit ahhhh right. you can still rebase against my repo then, i think looking for details try git rebase -i upstream (also, i’ve temporarily bumped bridgy user pages to show up to 100 responses. see if that gives you enough?)

E.g. Hey all...I have a question on Micropub, git and jekyll. Here is the situation I am working on: - I have a small server at Digital Ocean with Ubuntu 18.10, ruby and jekyll installed - I have a direct git connection between local and server. So no github, gitlab or any of that. - Based on instructions, I use a bare git repo on the server. So only git objects, refs etc. - When I git push from local, a post-receive script starts to run to copy the git repo and jekyll build it [...] - This all works fine when I post from my local machine to the server. - But when I want to use a micropub client, how do I deal with this? [...]

“Defining IndieWeb”

Two topics were grouped into a category called “Defining IndieWeb.” This topic group is characterized by discussions about IndieWeb’s principles and approaches, as well as comparison and contrasts with non-IndieWeb endeavours. The following topics were included in this group: 5, 13

As seen in Figure 5.4, conversations that reference IndieWeb’s principles are concentrated in these topics. I have argued in Chapter 4 that IndieWeb’s principles represent shared commitments about desirable outcomes from building for the IndieWeb, as well as prescriptions that guide processes during building. Accordingly, this topic is of particular importance for this dissertation.

Topic 5: Defining IndieWeb 1. This topic includes the most specific references to the *principles* page on IndieWeb’s wiki. Present in this topic are many conversations about IndieWeb’s goals and rationales for IndieWeb’s approach.

E.g. but for me the principles seem to be very similar to most open source or open web communities. I always thought of indieweb as the Open Web with adherence to very specific

protocols, that is probably wrong, but that's the impression for those of us on the edges of the community [...]

*E.g. we're deliberately avoiding that outcome by encouraging a plurality of projects. "one thing at its core" - those are the principles, rather than a github repo that's the difference [...] it's not just me that doesn't agree with the hateful language coming out of [non-IndieWeb project] related things those of us that started working on the indieweb did so more to be constructive and productive ourselves, rather than tear down big companies and spread hate We started with POSSE as a key *distinguishing* building block - to *use* existing networks rather than just hate against them."²*

Topic 13: Defining IndieWeb 2. Many of the most strongly representative documents labelled with this topic are parts of debates, including sharp criticisms of non IndieWeb-projects (though such comparison is present in topic 5 as well).

E.g. nah - RDF complicates the model unnecessarily with basing it on "triples" <http://microformats.org/wiki/triples> there's no "pretty much" about it more like "ugly much" nope - such challenges are based on tons of experience yes takes longer to explain = unnecessarily complicated anyway - no interest in arguing about RDF because that's a useless waste of time[...]

E.g. [...] <http://en.wiktionary.org/wiki/bikeshedding> (typically renaming things (arguing about) rather than actually doing anything meaningful) so this: "RSS is all about interoperability" is false. RSS is all about errantly focusing on plumbing instead of user experience.

“Events”

Topic 2 was labelled as “events”. This topic was the most straightforward topic to classify, since it is characterized by terms such as “event”, “hwc” (an acronym for “Homebrew Website Club”), “iwc” (an acronym for “IndieWebCamp”), and “venue,” which make it easy to infer that this topic refers to Events. Several other topics less immediately clear. Additionally, Figure 5.7 shows a high concentration of event-related keywords in topic 2.

E.g. How is September for IWC NYC? NYC or MIT would be good for September, very likely we can at least do one then SF near TPAC (either in late Oct or early Nov) and LA afterwards, since its still warm in LA in late Nov / early Dec [...]

E.g. good evening #indiewebcamp! totally fair [name] - happy holidays! good evening [name] and good news! I (finally) visited the Mozilla NYC (Manhattan) offices tonight correct and suffice it to say, there is space to do a 8-15 person IWC NYC on a weekend

²Since this excerpt includes a strong claim about a non-IndieWeb project but does not include context for that statement, I have removed the name of the non-IndieWeb project that was being contrasted with IndieWeb's approach.

“Online community management”

Topic 14 was labelled as “Online community management.” This topic consists mostly of discussions related to IndieWeb’s chat channels, wiki, and other aspects of online community. Many of the most representative documents in this topic were related to online support for events, such as setting up video streams or other online utilities for IndieWebCamps. However, closer inspection identified non-event conversations concerning IndieWeb’s wiki and other topics. Further, Figure 5.7 shows that event-related keywords were not represented in topic 14, affirming that this is semantically distinct from topic 2 (*events*).

E.g. [...] *Do we have all the stream and hangout links to add to the schedule? [name]? [name]? [name]? All hashtags with linked etherpads are on the schedule [name]: all live stream links are up Will add your hangouts link! Was hoping to get hangout links for the other rooms [...]*

E.g. *“For some reason the Jibri server that worked perfectly fine for recording Jitsi two weeks ago, today doesn’t want to work. I’ve been fighting with it for the last hour [...]*

“Other topics”

Topics 1, 3, and 6 were categorized as ‘other.’ As discussed in Section 5.2.1, topic 1 had a low topic probability even when it was classified as a document’s dominant topic. Additionally, documents classified with topic 1 as their dominant topic demonstrated poor similarity.

Several of the most representative documents for topic 3 could be identified as referring to IndieWeb’s chat channels, as in the following examples:

E.g. [...] *this channel is not just slack, but IRC. Loqi as an irc bot (type “who is Loqi?”) which handles commands, like leaving messages for other users. So when you type “!tell xyz”, Loqi picks it up and when user “xyz” returns - types something -, Loqi plays the message back.*

E.g. [...] *We’ve had a lot of spam issues with IRC [...] You are still in the slack channel but your connection to IRC is a little broken currently because of our spam detection was turned up to [100] Let me see if we can get the IRC channel back open for you [...]*

However, when reviewing a random selection of topic 3’s documents, it became clear that most did not have this focus, such as the following example:

E.g. *and the answer is not only did it work, (setting up an iPod 6 with iOS 11 from a backup from an iPod 5 touch with iOS 9) but it specifically did *not* reinstall all the apps from backup (or the apps were not part of the backup, possibly?) so now I’m reinstalling only apps I use every day a few at a time to minimize new apps for the new year or apps at all that is after I installed Gmail, the wifi got flakey.*

Similarly, I failed to identify a consistent area of discussion when reviewing documents from topic 6. Documents classified with this topic generally related to technical topics, but without sufficient consistency to clearly identify a suitable label.

E.g. everything is hackable. thats like virusses for microsoft pc’s. to be honest, this is the side effect of DIY ... it’s obviously not DIY. Or not yet enough. Lately I get every month help-requests from DIY people that are stuck

E.g. Twilio is lovely and well worth playing with I may implement Twilio into Ferocity so I can text posts in Twilio just uses web hooks - so every time you get a message sent to your number, it sends an HTTP POST message to a specified endpoint

This “other” group is preserved in subsequent stages of analysis. However, the contents of this group were not identified as meaningful in the context of this project’s research questions

5.2.4 Results: cluster analysis of topic trends over time

Once topics were labelled, a summary was created of each individuals’ chat activity per month following the procedure described in Section 3.3.4.

The results of clustering topic shares are presented here. The clustering algorithm first produces a dendrogram showing the hierarchy of clusters, which is shown in Figure 5.8. The dendrogram shows one cluster of 3092 that is clearly distinct from the others, and the remainder of observations are defined less dramatically. Two types of visualizations were used to evaluate and label the clusters, displayed in Figure 5.9 and Figure 5.10. Multiple cluster counts were tested, and I found that seven clusters led to the most meaningful results. The red line on Figure 5.8 indicates the height cutoff value at which seven clusters were generated.

Figure 5.9 displays a radar plot showing the distribution of normalized topic shares among each cluster. Every tenth percentile is displayed with 10% opacity, so solid blue areas of the plot display lower percentile values and pale areas display outliers. Figure 5.10 uses violin charts to present a more detailed view of this same distribution. Both visualizations were used to explore this data, since Figure 5.9 presents an overview that could be assessed quickly, and Figure 5.10 presents a more detailed perspective. These charts show that the seven clusters each represent significantly different balances of discussion devoted to each cluster. The largest cluster of 3092 observations represents observations where a very small proportion of discussions were labelled with topics. Apart from the large cluster of 3092 observations, each cluster is dominated by a single topic. This indicates that there is not a widespread tendency for individuals to blend the same combinations of topics in a given month. For example, there was no consistent pattern in which people who engage in conversations about “Events” were also likely to discuss “Community management” in equal measure during the same month. Based on this figure, the clusters have been labelled using the same titles as the topic groups they most strongly represent:

- **Cluster 0:** IndieWeb building blocks
- **Cluster 1:** Web dev (general)
- **Cluster 2:** Defining IndieWeb
- **Cluster 3:** No topic identified
- **Cluster 4:** Events
- **Cluster 5:** Other topic

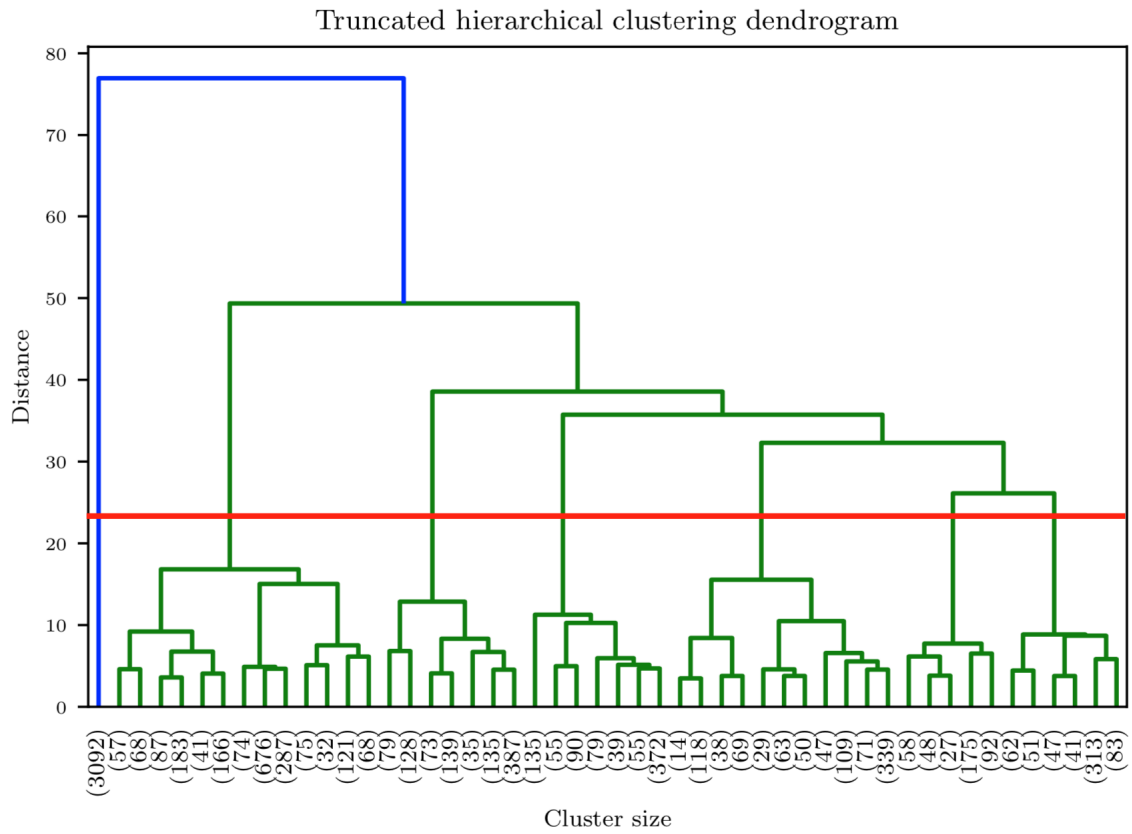


Figure 5.8: Hierarchical dendrogram of topic clustering results. The red line indicates the cutoff point used to determine the number of clusters (7).

- **Cluster 6: Community management**

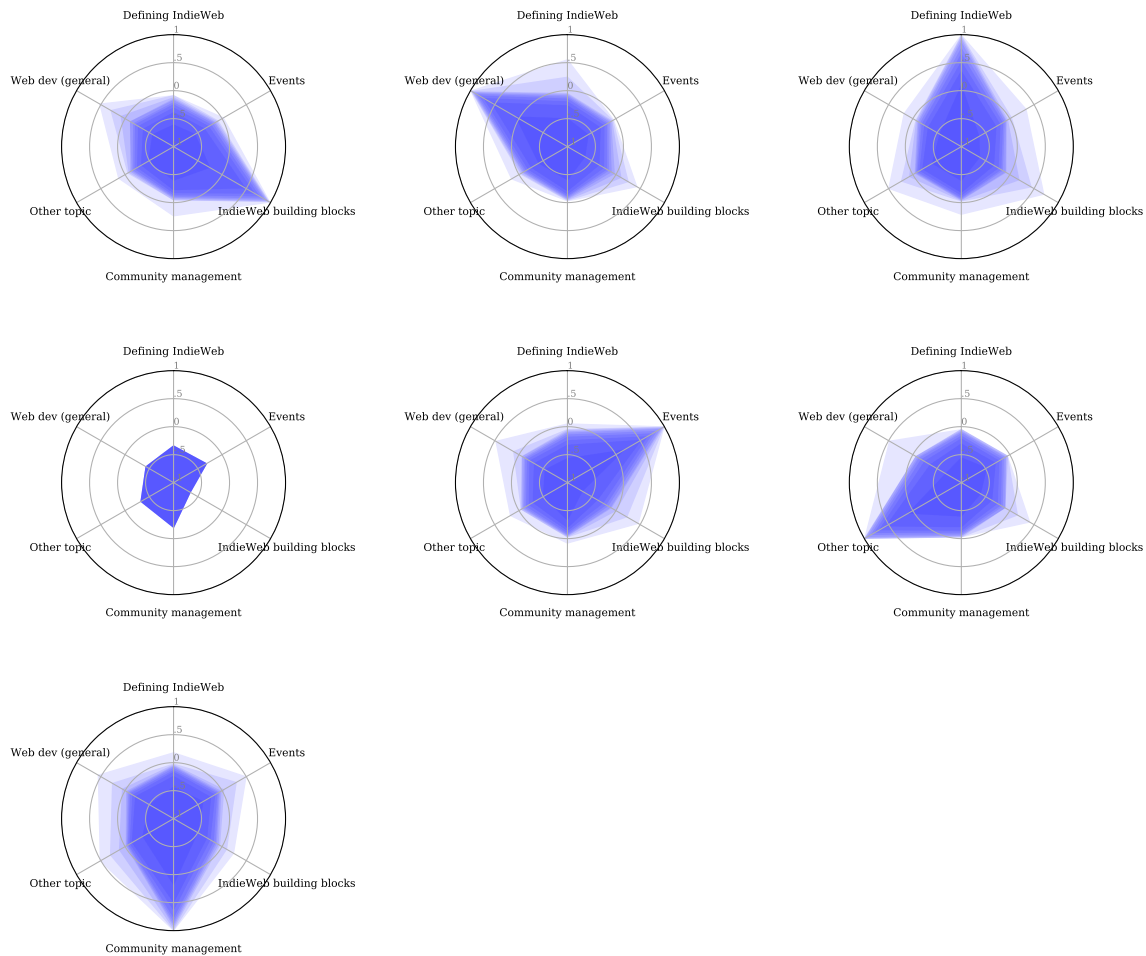


Figure 5.9: Radar plots showing weight of normalized topic shares per cluster. Every tenth percentile is overlaid on top of one another in transparent blue. Light (transparent) shades indicate values at high percentiles, and darker (more opaque) shades indicate values at lower percentiles.

5.2.5 Topics over time

Based on the results of the topic cluster analysis, a stackplot chart was generated showing the proportion of observations associated with each topic over time. This stackplot is presented in Figure 5.11. This figure charts the trend as calculated by the seasonal decomposition function in *statsmodels*.³ Appendix I includes an unsmoothed version.

Several patterns are evident in this chart. One of the strongest patterns is that the proportion of observations for which no dominant topic could be identified, indicated in pink, decreases dramatically from the beginning of the chart in mid 2011 to 2014. A possible explanation for this is that there was less activity on IndieWeb’s chat channels during this period, and a lower density of text made it difficult for the LDA model to reliably identify topics. Additionally, several topics were defined by their

³See https://www.statsmodels.org/dev/generated/statsmodels.tsa.seasonal.seasonal_decompose.html

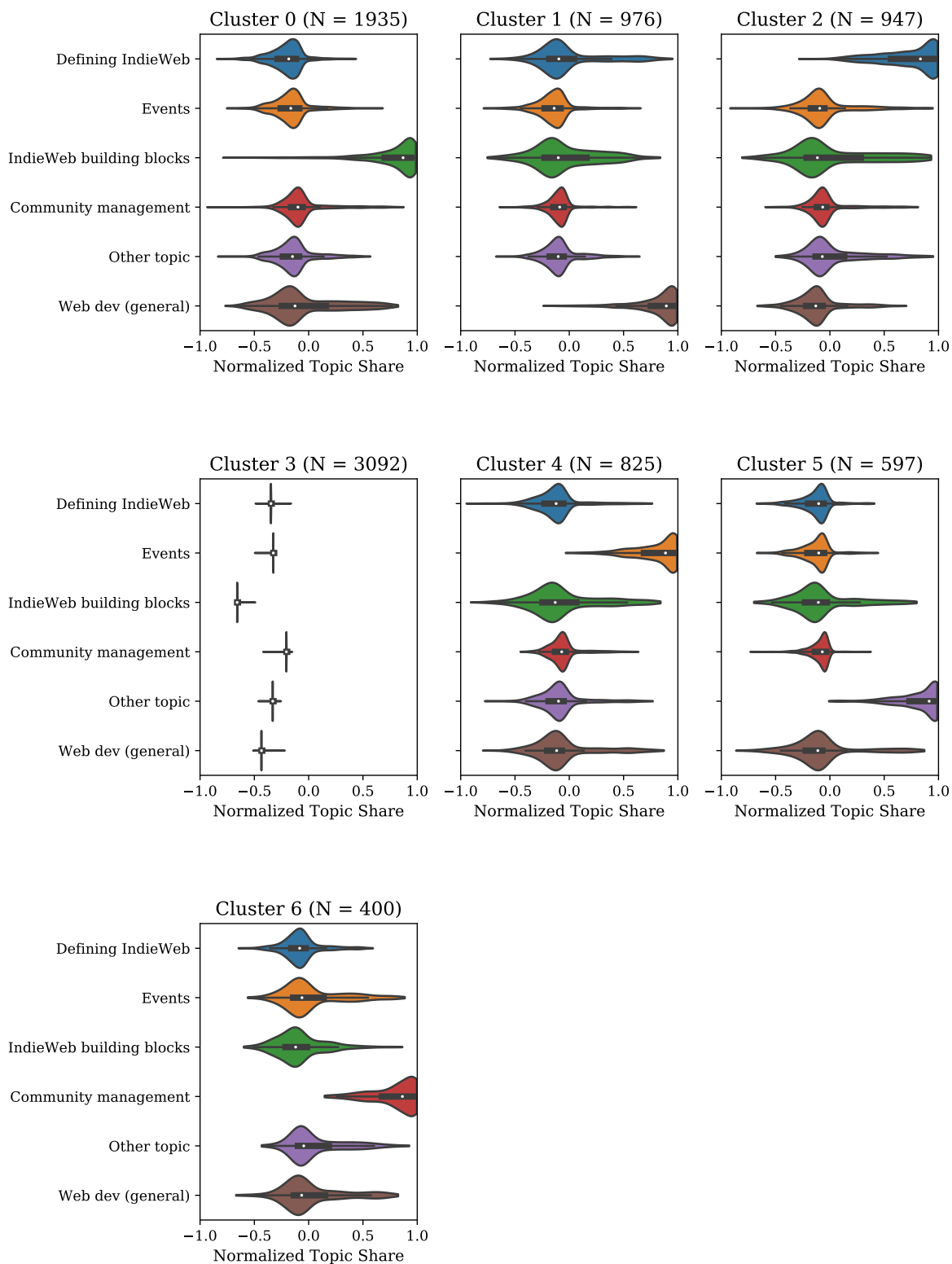


Figure 5.10: Violin charts showing distribution of topic shares in each cluster. X-axis shows topic share normalized to a range of -1.00 to 1.00.

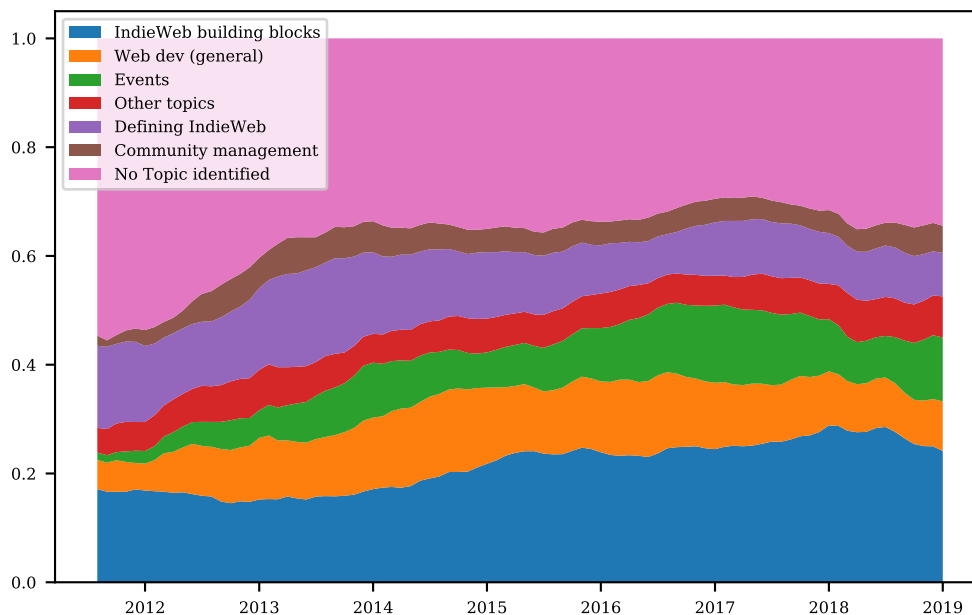


Figure 5.11: Proportion of chat clusters related to each topic over time.

inclusion of specific terms to describe IndieWeb building blocks, event names, and other concepts that were emerging in the first few years of IndieWeb’s history. Many of these terms were created during the period represented by this data, and so were absent from earlier logs. As a result, earlier logs were likely a poor fit for the topic model since they lack those terms.

The proportion of observations associated with *IndieWeb building blocks* increases over time. In part, these are conversations about building for the IndieWeb, but also include conversations about using IndieWeb software made by others. In both instances, this suggests that the focus of discussions has increasingly focused on IndieWeb’s technical features over time. The proportion of observations associated with the *Defining IndieWeb* topic decreases over time, though remains significant. A likely explanation for this trend downward is that it was necessary to define and defend IndieWeb’s approach in its earlier days, and that this definition has stabilized over time. The proportion of observations associated with the *Events* topic increases fairly steadily, indicating that events have been an increasing priority among IndieWeb’s chat participants. The proportion of observations classified as being related to *community management* was very low at the beginning of the archive, then grew from 2011-2013 from which point it has remained fairly steady.

5.3 Summary statistics about IndieWeb’s GitHub network

Using the procedure described in Appendix D, 1,102 IndieWeb-related repositories were identified. Of these, 463 repositories were forks, and 639 were not. A fork is a copy of a repository, in which one can make changes without affecting the original. As mentioned in Chapter 4, forks can either be used to build upon an existing project for one’s own purposes, or the changes may be submitted back to the original repository as a pull request.

Because forks are duplicates of an original repository, they duplicate the repository’s log of commits

(however, events related to ‘issues’ are not copied). 355 Repositories that did not contain any unique events—i.e. where a repository was forked but never built upon—were removed from this analysis, leaving 747 repositories remaining.

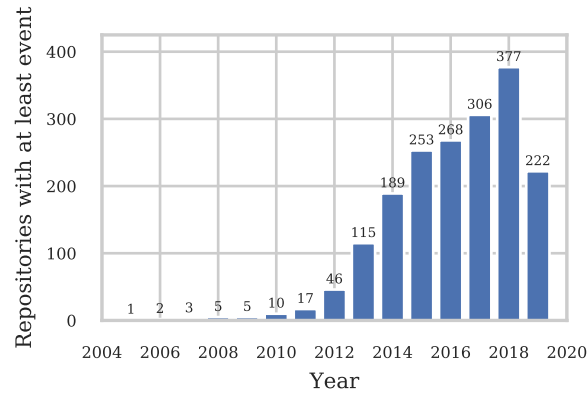


Figure 5.12: IndieWeb-related GitHub repositories with at least one event by year.

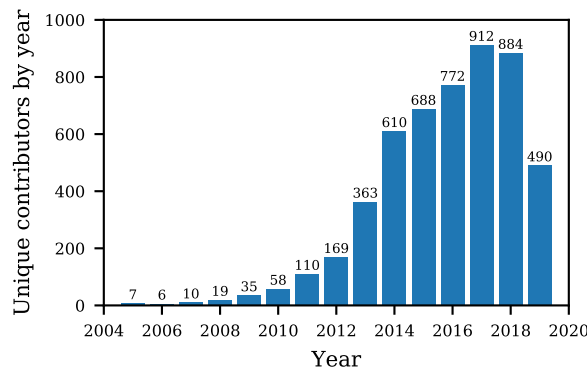


Figure 5.13: People who made at least one contribution to IndieWeb-related repositories by year.

Figure 5.12 shows the number of repositories that had at least one event by year, and Figure 5.13 shows the number of individuals who made a contribution to one of the identified repositories, by year. These charts show a steadily increasing number of repositories and contributing users each year. There appears to be a decrease in repositories/contributors in 2019 because the data represented in these charts was downloaded on May 17, 2019.

An additional consideration for interpreting these figures is that, although each of the repositories included in this data has been identified as related to IndieWeb using the procedure in Appendix D, some of these repositories are only tangentially related. This is why the data includes activity from as early as 2005, before IndieWeb was founded. This indicates that repositories that predate IndieWeb were at some point modified to include references to IndieWeb or IndieWeb-related keywords.

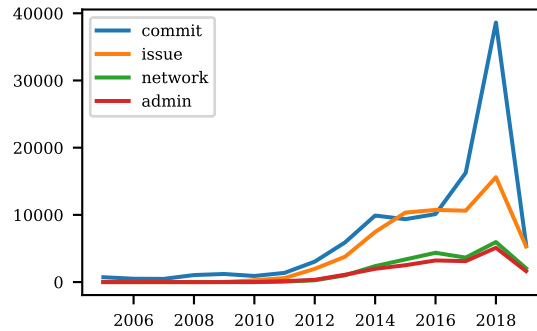


Figure 5.14: Github actions by year.

5.3.1 Event types by year

As described in Section 3.3.3, events from these repos were gathered and organized into a taxonomy of four action types:

- *commits*: Changes to code or files.
- *issue events*: Opening or commenting upon issues.
- *network events*: Linking related issues and/or commits to one another.
- *admin events*: Marking issues as complete or assigning labels to issues.

Figure 5.14 displays a count of events for each year in the dataset, grouped by action type. This chart shows a growth pattern that mirrors the number of repositories in Figure 5.12 and the number of users in Figure 5.13. Commits are generally greater in number than other action types, with the exception of an increase in the number of issue events and a decrease in the number of commits in late 2014, which contributes to a slightly higher number of issue events than commits through 2015. A possible reason for the growth in the proportion of issue-actions to commits is that, as seen in Figures 5.1 and 5.2, IndieWeb’s chat grew substantially in 2014, and IndieWeb generally increased in profile during this time. This would have contributed to a higher number of users, which could increase the amount of bug reports and feature requests posted to GitHub.

5.3.2 Distribution of commits and issues among repos

Figure 5.15 shows the distribution of user interactions by repository, organized by event type. The median number of users who contributed a commit to a repository was 1, compared to 3 users who performed issue events, 1 who performed admin events, and 2 who performed network events. This shows that the majority of GitHub repositories captured in this analysis had only one person committing code. The higher median for issue and network events suggests that many projects had multiple users who contributed by testing or offering feedback.

Figure 5.16 presents a network chart of GitHub activity during 2019. Appendix J contains charts from other years and a chart representing GitHub activity from 2011-May 17, 2019. Repos are colour coded based on the prominence at which they include the IndieWeb keywords defined in Step 3 of the procedure in Appendix D. Repos that contain an IndieWeb-related keyword in the repository name or

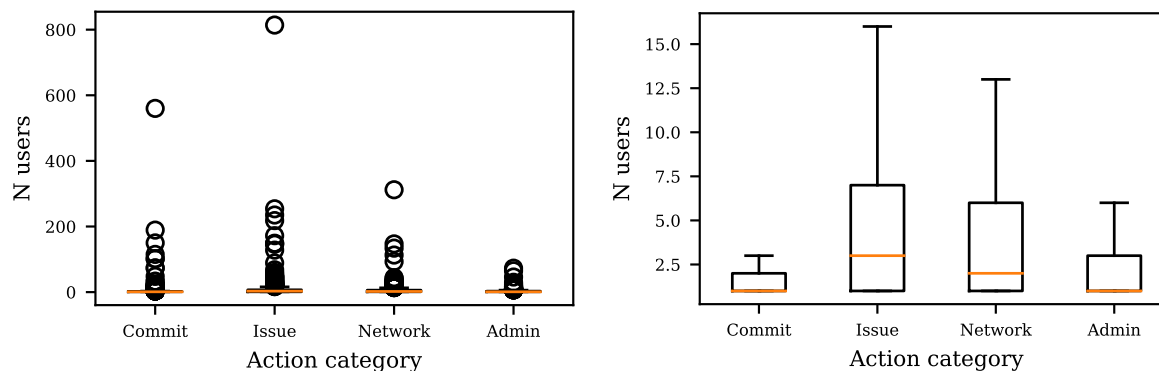


Figure 5.15: Distribution of users per repository by action category. *Left:* Including Outliers. *Right:* Outliers removed.

description are red. Repos that only contain one of the keywords in their README file are coloured blue. This is used as a general indicator of how central IndieWeb concepts are to the Repository. For example, the list of repositories includes examples such as *Publify*, whose README includes the following:

Publify is a simple but full featured web publishing software. It’s built around a blogging engine and a small message system connected to Twitter.

Publify follows the principles of the IndieWeb, which are self hosting your Web site, and Publish On your Own Site, Syndicate Everywhere.

Publify has been around since 2004 and is the oldest Ruby on Rails open source project alive. (<https://github.com/publify/publify/>, quoted on May 24, 2020)

Publify is obviously IndieWeb-related, evidenced by its reference to IndieWeb principles. However, it also predates IndieWeb, does not support core IndieWeb features such as Webmentions (though Webmention support is reported as a forthcoming feature), and the *Publify* article on IndieWeb’s wiki is brief and has not been updated since 2016. By contrast, a repository such as <https://github.com/indieweb/wordpress-indieauth> contains the term “IndieAuth” in its title, indicating it is centrally focused on an IndieWeb technology.

Figure 5.16 illustrates a trend that the repositories with the largest number of contributors are, by the metric explained above, less strongly related to the IndieWeb than average. To verify this assessment, a Mann-Whitney U-test was performed, measuring degree centrality among strongly and weakly IndieWeb-related repositories. The test was performed on a network graph representing activity from Jan 1, 2011 to May 17, 2019. The network contained one subgraph of 2611 nodes (including both users and repos) and 20 smaller subgraphs with a median of 2 nodes each. The smaller subgraphs were dropped from the analysis, leaving 729 repositories from the larger subgraph. Of those, 61% (N=444) were strongly IndieWeb-related and 39% (N=285) were weakly IndieWeb-related. The Mann-Whitney U-test indicated that the degree centrality (indicating the number of contributing users) was greater for less IndieWeb-focused repos (Mdn = .0007666) than for more IndieWeb-focused repos (Mdn = .0003833), $U = 56213$, $p = 0.0055$. Full Mann-Whitney U-test results are presented in Appendix K.1. This shows that more strongly IndieWeb-focused repositories are contributed to by a smaller number of people. This supports the observation in Chapter 4 that IndieWeb projects are largely built by individuals or small

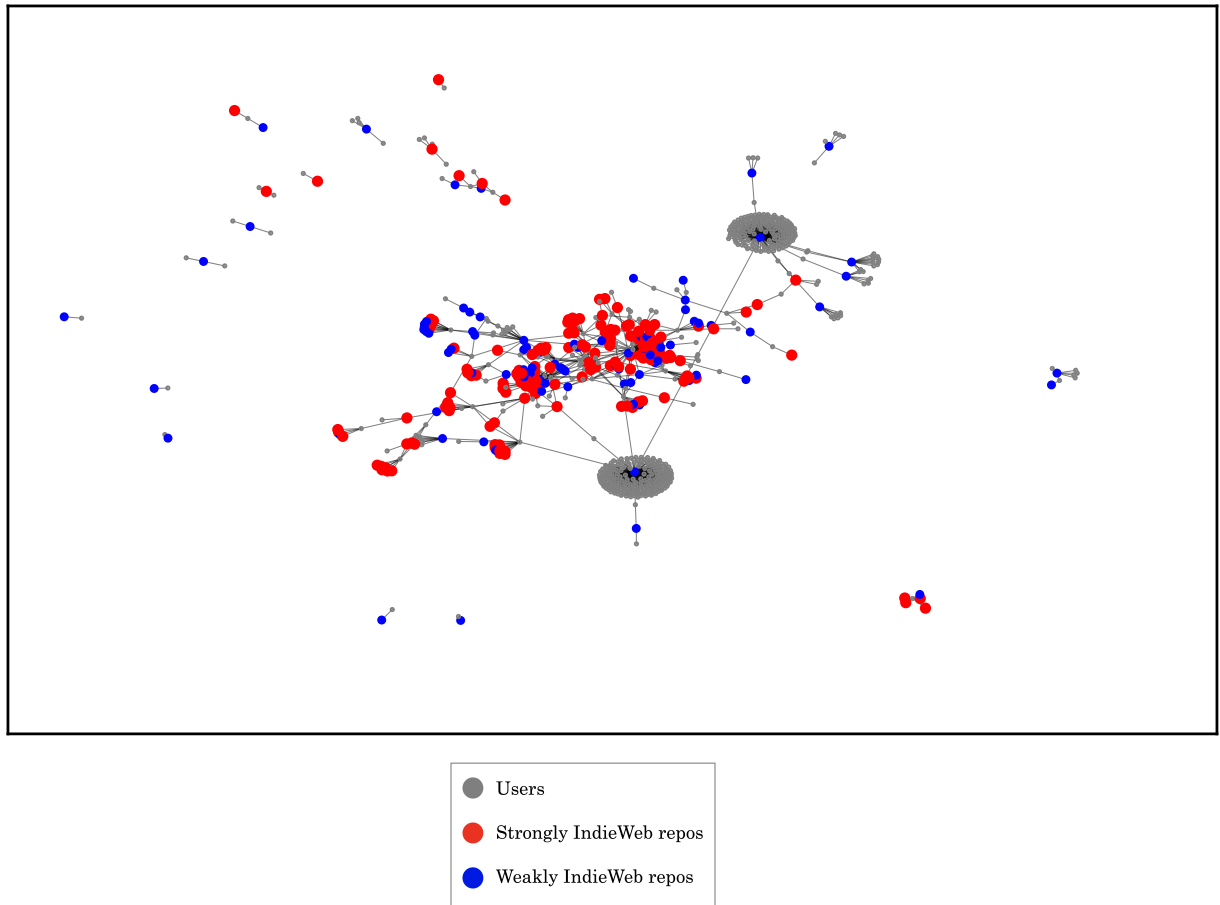


Figure 5.16: Network chart of IndieWeb-related GitHub repos and users from Jan 1, 2019 through May 17, 2019.

groups, and that collaboration occurs through relationships of interoperability and dependence among these projects.

5.4 Does chat activity relate to GitHub activity?

The previous sections have identified structures of centralization and decentralization in IndieWeb’s chat and network of GitHub repositories. The extent to which participation in IndieWeb’s chat may be connected to activity on GitHub is an important consideration for mapping influence. Do centres of power in IndieWeb’s chat predict corresponding centres on GitHub? To address this question, information about people who had activity on both IndieWeb’s chat channels and on GitHub was linked where possible. This section describes the process of that data linkage,

5.4.1 Linking between GitHub and chat

Of 2,896 uniquely identified users on IndieWeb’s chat, 105 (3.6%) were linked to a GitHub account which had interacted with the IndieWeb-related GitHub repositories included in this analysis. In this section, I identify features of users’ chat characteristics that indicate likelihood to be linked to GitHub activity. Firstly, it is important to acknowledge that this is not likely to be a measurement of all participation in GitHub repositories related to IndieWeb, but rather a measurement of participation that could be linked using the methods described in Section 3.3.4. The 3.6% of chat users who were linked to activity on GitHub are those who (a) interacted with a GitHub repository that was captured in this study, (b) registered a personal URL on IndieWeb’s list of chat-names⁴, and (c) linked to their personal URL from their GitHub account, or to their GitHub account from their personal URL.

Table 5.4 shows the result of a logistic regression predicting to identified GitHub activity among IndieWeb chat participants. The dependent variable is a Boolean indicator of whether the user had any GitHub activity with the IndieWeb-related repositories captured by this analysis. Model 1 tests using two independent variables, (1) the number of months during which each user posted at least one message to IndieWeb’s chat, and (2) the total number of chat messages sent by each user. Model 2 tests using (1) the number of months during which each user posted at least one message to IndieWeb’s chat and (2) variables presenting the number of months each topic was dominant across the users’ chat activity. The number of chat months was excluded from model 2 because it is collinear with the counts of months for each topic.

Model 1 shows that long-time participants on IndieWeb’s chat were more likely be identified as interacting with IndieWeb-related Github repositories. Model 2 shows that regularly posting about specific topics was predictive of being active on GitHub. Specifically, a large number of months during which one’s chat messages were classified as being about either *IndieWeb building blocks* or *Events* was associated with a greater likelihood of identified GitHub participation. By contrast, the number of months in which a users’ chat messages focused on other topics was not predictive of GitHub activity. That discussions related to IndieWeb building blocks are predictive of GitHub activity is expected, since conversations related to IndieWeb-related development (which is typically hosted on GitHub) are likely to use language related to IndieWeb technologies. That discussions related to events are predictive of GitHub activity supports the observation in Chapter 4 that IndieWeb events are oriented around

⁴<https://indieweb.org/chat-names>

building. Further, attending in-person events (and as a consequence, posting about them on IndieWeb's chat) demonstrates an investment of time and energy toward the IndieWeb, and it is unsurprising that such an investment is predictive of using GitHub to interact with IndieWeb projects, as well as performing the steps necessary for my analysis to successfully cross-reference one's chat username with a GitHub profile.

Table 5.4: Logistic Regression predicting to identified GitHub activity among IndieWeb chat participants

	Model 1	Model 2
N chat months	0.197** (12.50)	
N messages	0.0969 (0.32)	-0.370 (-1.25)
N months per topic cluster:		
Other topic		-0.0297 (-0.18)
No topic		0.00250 (0.07)
IndieWeb building blocks		0.584*** (9.28)
Online community management		-0.0117 (-0.06)
Defining IndieWeb		0.232 (1.73)
Events		0.458*** (4.79)
Web dev (general)		0.0997 (1.09)
Constant	-4.820*** (-25.07)	-5.032*** (-22.86)
Observations	2896	2896
Pseudo R^2	0.494	0.584

t statistics in parentheses

* $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$

5.4.2 Cluster analysis of GitHub activities

The next step in this analysis was to investigate whether chat activity is predictive of specific types of GitHub activity. Similar to the previously described analysis of chat topics, a cluster analysis was used to group observations according to their count of GitHub events by type. A dendrogram in Figure 5.17 shows that two clusters can explain much of the variance among counts of the four types of GitHub events. As with topics, radar and violin plots were used to visualize the distribution of variables among these two clusters. These charts are presented in Figures 5.18 and 5.19. As an additional check to confirm

that variance among the four types of GitHub events could be suitably represented by only *commits* and *issue events*, Figure 5.20 show a scatter plot comparing *commit* and *issue event* scores among the two clusters. This figure illustrates that the observations with a high proportion of commits compared to other actions were neatly clustered apart from those with a high proportion of issue events. Based on these figures, observations that included GitHub activity were labelled as either *commit-focused* or *issue-focused*. Since it was possible for observations to describe cases where an individual had posted to IndieWeb's chat but had not interacted on GitHub in a given month, observations that were not labelled as either *commit-focused* or *issue-focused* indicate that there was no GitHub activity.

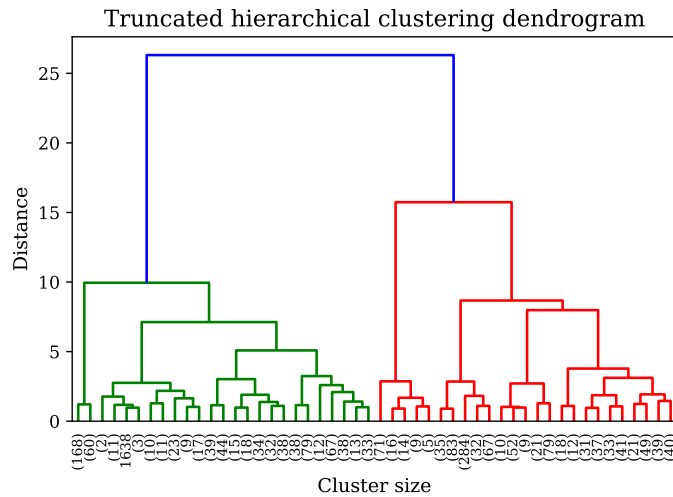


Figure 5.17: Hierarchical dendrogram of GitHub activity clustering results.

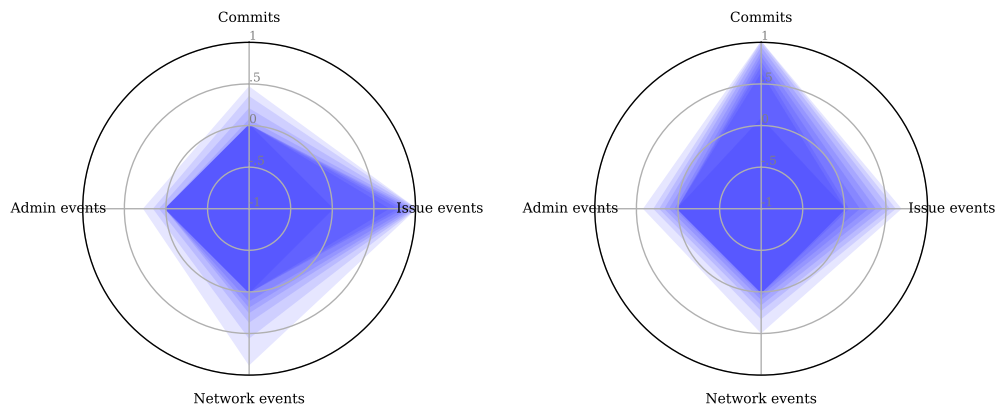


Figure 5.18: Radar plots showing weight of GitHub actions per cluster. Every tenth percentile is overlaid on top of one another in transparent blue. Light (transparent) shades indicate values at high percentiles, and darker (more opaque) shades indicate values at lower percentiles.

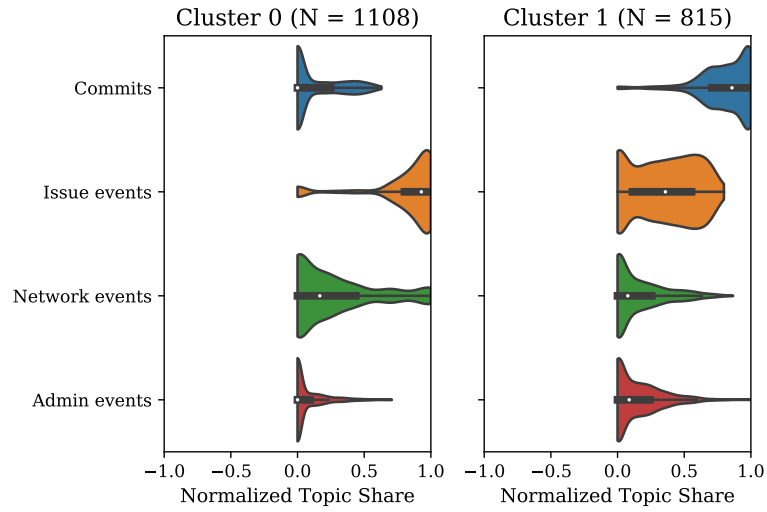


Figure 5.19: Violin charts showing distribution of topic group values in each cluster. X-axis shows topic share normalized to a range of 0.00 to 1.00.

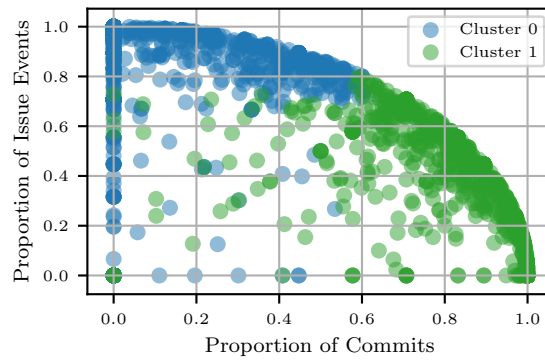


Figure 5.20: Scatter plot showing distribution of commit and issue scores among two clusters.

5.4.3 Connecting chat activity to commit-focused and issue-focused GitHub activity

Having clustered observations by both topic and type of GitHub activity, logistic regression analyses were performed predicting to Boolean variables indicating whether a user was clustered as being *commit-focused* or *issue-focused* in a given month. The independent variables are (1) The topic cluster assigned to each observation, with 'no topic assigned' excluded as a dummy category (2) the number of messages posted, and (3) the betweenness centrality of the user in IndieWeb's chat that month. The results of these analyses are presented in Tables 5.5 and 5.6. Unlike the analysis in Table 5.4, these regressions investigate activity over time. Accordingly, the regressions in 5.5 and 5.6 present a separate model for each year as well as an overall model that includes all years. 2011 was excluded since there are not enough GitHub events. Both tables include pseudo R2 scores indicating their models' goodness-of-fit. The regression was performed using the `logit` command in *Stata/IC 15.1*, which reports McFadden's pseudo R2. McFadden (1978) explains that pseudo R2 scores are generally lower than R2, and values of 0.2-0.4 "represent an excellent fit" (p. 307). Neither table consistently meets this standard, indicating that there is substantial unexplained variance. However, neither do they consistently fall way short of this standard. Accordingly, I interpret these regression results as a useful predictor of GitHub activity, albeit an incomplete one.

Model 9 of both tables tested data for all years from 2012-2019. For both commit-focused and issue-focused activity, all of the independent variables were significant predictors. That all topic clusters were significant indicates that, overall, in a given month people whose chat messages were "on-topic" enough to be classified by the topic model were more likely to active on GitHub during that same period. Similarly, that both the number of messages posted and a user's betweenness centrality were significant indicates that generally being active on chat was a predictor of both kinds of GitHub activity.

A more detailed interpretation can be achieved by examining the yearly Models (1 through 8). In both tables, the number of messages posted is a significant predictor of GitHub activity in almost every year. By contrast, betweenness centrality is inconsistently significant across years. This suggests that investing time to post a lot of messages predicts GitHub activity moreso than the number of other chat participants who responded to one's messages.

With regard to topic clusters, both tables demonstrate similar patterns, with "IndieWeb building blocks", "Events" (from 2014-2015 onward), and "Web dev (general)" being consistent predictors of activity. The main difference between these tables is that "Defining IndieWeb" is only a significant predictor of commit-focused GitHub activity in 2015 and 2018 but is a significant predictor of issue-focused GitHub activity in 2014-2018. Similarly, comparing Model 9 of both tables shows that "Defining IndieWeb," though significant, has a relatively low coefficient for predicting commit-focused GitHub activity (0.876) compared to a more average coefficient for predicting issue-focused GitHub activity (1.336). "Online community management" was a predictor for commit-focused activity from 2013-2017, but a predictor for issue-focused activity in 2014-2015 and 2018-2019. This could indicate that those who discuss online community management have become more issue-focused in the 2018-2019, but this would require further investigation to assess.

Finally, "Other topics" was only significant in two years for both types of activity, indicating that overall it is not a significant predictor of GitHub activity when compared to other topics.

In sum, these regression results show that people who post a lot of messages to chat are more likely to be active on IndieWeb-related GitHub repositories. Development-focused chat messages, such as those

relating to “IndieWeb building blocks” and “Web dev (general),” and chat messages related to “events” were consistent predictors of both types of activities over time. “Defining IndieWeb was a stronger predictor for issue-focused than commit-focused GitHub activity.

Table 5.5: Logistic Regression predicting to *commit-focused* GitHub activity

	Model 1 2012	Model 2 2013	Model 3 2014	Model 4 2015	Model 5 2016	Model 6 2017	Model 7 2018	Model 8 2019	Model 9 All years
Topic clusters:									
Other Topic	0 (.)	0.319 (0.28)	0.987 (1.37)	1.354 (1.47)	0.929 (1.50)	1.240** (3.28)	1.823*** (4.91)	0.160 (0.20)	1.310*** (6.56)
IndieWeb building blocks	2.271* (2.00)	2.332*** (4.07)	2.358*** (5.22)	3.073*** (5.07)	1.978*** (5.21)	1.695*** (6.16)	2.200*** (7.07)	1.177** (2.59)	2.077*** (14.56)
Online community management	1.925 (1.43)	1.897** (2.69)	1.618** (2.58)	2.442** (3.22)	1.521** (2.74)	1.711*** (4.12)	1.019 (1.73)	0.986 (1.66)	1.576*** (7.73)
Defining IndieWeb	1.211 (0.91)	0.964 (1.47)	0.408 (0.67)	1.802* (2.53)	0.557 (0.95)	0.372 (0.91)	1.257** (3.07)	0.873 (1.46)	0.876*** (4.63)
Events	0 (.)	1.073 (1.36)	1.833*** (3.43)	3.144*** (4.84)	2.038*** (5.01)	0.953** (2.90)	2.076*** (5.44)	1.527** (3.24)	1.828*** (11.10)
Web dev (general)	2.551* (2.19)	1.996** (3.28)	1.569** (3.20)	2.538*** (3.90)	1.686*** (4.09)	1.245*** (3.73)	2.396*** (6.89)	1.328* (2.53)	1.803*** (11.29)
N messages	0.210*** (3.35)	0.0529*** (3.87)	0.0526*** (5.68)	0.0553*** (5.02)	0.0486*** (3.77)	0.0251* (2.18)	0.0476*** (4.79)	0.0556*** (3.38)	0.0472*** (11.44)
Chat betweenness	0.0155* (2.32)	0.0160** (3.01)	0.0145** (3.00)	0.00111 (0.25)	0.000288 (0.06)	0.00796* (2.38)	0.00145 (0.38)	0.00425 (0.88)	0.00700*** (4.80)
Constant	-5.143*** (-5.11)	-4.169*** (-8.26)	-4.331*** (-10.54)	-4.911*** (-8.48)	-3.646*** (-10.79)	-2.930*** (-12.43)	-3.967*** (-14.17)	-3.642*** (-9.51)	-3.890*** (-30.55)
Observations	304	712	1358	1149	1102	1223	1889	902	8923
Pseudo R^2	0.340	0.200	0.193	0.181	0.100	0.084	0.124	0.077	0.122

t statistics in parentheses

* $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$

Table 5.6: Logistic Regression predicting to *issue-focused* GitHub activity

	Model 1 2012	Model 2 2013	Model 3 2014	Model 4 2015	Model 5 2016	Model 6 2017	Model 7 2018	Model 8 2019	Model 9 All years
Topic clusters:									
Other Topic	16.45 (0.00)	0 (.)	1.658 (1.17)	1.766** (3.19)	0.874 (1.83)	0.639 (1.48)	0.456 (1.33)	1.052* (2.19)	0.957*** (5.07)
IndieWeb building blocks	15.31 (0.00)	2.304* (2.10)	3.513*** (3.42)	1.891*** (4.41)	1.571*** (5.16)	1.726*** (6.15)	2.245*** (10.49)	1.824*** (5.38)	2.074*** (16.70)
Online community management	0 (.)	1.574 (1.10)	2.429* (1.96)	1.429* (2.26)	0.705 (1.33)	0.290 (0.49)	1.525*** (4.31)	1.517*** (3.44)	1.283*** (6.78)
Defining IndieWeb	13.13 (0.00)	1.821 (1.60)	3.160** (3.01)	1.810*** (3.74)	1.078** (2.63)	1.297*** (3.74)	1.729*** (6.53)	0.653 (1.33)	1.336*** (8.85)
Events	0 (.)	2.134 (1.73)	1.953 (1.58)	1.764*** (3.44)	1.718*** (5.21)	2.007*** (6.70)	1.247*** (4.08)	1.548*** (4.17)	1.897*** (13.32)
Web dev (general)	0 (.)	2.267* (2.00)	3.147** (3.02)	1.962*** (4.23)	1.783*** (5.58)	1.506*** (4.64)	1.650*** (6.16)	0.909* (2.03)	1.647*** (11.64)
N messages	0.212 (1.81)	0.0714*** (4.42)	0.0598*** (6.08)	0.0663*** (5.53)	0.109*** (6.15)	0.0966*** (6.40)	0.0825*** (6.70)	0.0897*** (5.57)	0.0723*** (15.63)
Chat betweenness	0.0164 (1.36)	0.00395 (0.53)	0.00947 (1.78)	0.00991* (2.38)	0.00445 (1.09)	0.0105** (3.10)	0.0132*** (3.85)	0.00635 (1.62)	0.00797*** (5.79)
Constant	-20.08 (-0.00)	-5.544*** (-5.53)	-6.127*** (-6.12)	-4.066*** (-10.66)	-3.049*** (-11.91)	-2.987*** (-12.35)	-3.083*** (-16.76)	-3.083*** (-10.44)	-3.569*** (-32.62)
Observations	277	669	1358	1149	1102	1223	1889	902	8923
Pseudo R^2	0.365	0.206	0.248	0.178	0.159	0.182	0.216	0.153	0.166

t statistics in parentheses

* $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$

5.5 Limitations

There are a number of limitations that result from the research methods used in this chapter. Most notably, the measures used to analyze IndieWeb's chat activities are based on quantities of messages, which renders it difficult to identify quantitatively small but qualitatively significant contributions.

Additionally, the method of identifying IndieWeb-related repositories on GitHub excludes projects that are not open-source as well as projects that are open source but hosted somewhere besides GitHub. Furthermore, the list of IndieWeb-related repositories was seeded by identifying repositories that were linked in IndieWeb's wiki. It is possible that this led to the under-representation of projects by more peripheral IndieWeb contributors, though an attempt has been made to address this by expanding the list of repositories across several steps (see Appendix D). One type of development activity likely to be under-represented is people's work on personal websites which are often not shared on GitHub.

I acknowledge these limitations as they pertain to this chapter and have attempted to mitigate them by combining this chapter's analysis with more ethnographic and qualitative analyses elsewhere in this dissertation.

5.6 Discussion and Conclusion

This chapter has structural features of IndieWeb's online community, including its chat discussions and network of IndieWeb-related GitHub repositories. This investigation begins to address **RQ3**: *Is power over planning and executing design activities centralized or decentralized among stakeholders, and is anyone included or excluded as a result?*

By analyzing publicly archived interactional data from I have identified evidence for a centre of influence in IndieWeb's chat, which was evident based on the distribution of individuals' (1) total chat messages, (2) monthly chat messages, and (3) number of active chat months. People in this centre are likely to have pronounced influence over the kind of discussions that take place in IndieWeb's chat, including shifts in the prevalence of topics over time, such as increasing attention toward the topics I have labelled *Events* and *IndieWeb building blocks*, and slightly less attention toward the topic I have labelled *Defining IndieWeb*. Further, I have identified that people who are at this centre are more likely to be contributing to IndieWeb-related repositories on GitHub than people with less chat participation. Moreover, participating in chat discussions related to the *Events* or *IndieWeb building blocks* was a predictor of GitHub activity related to both commits and issues, and participating in chat discussions about *Defining IndieWeb* were a predictor of issue-focused GitHub Activity. All of these patterns indicate that IndieWeb's community has a centre of very active participants who are influential over both chat discussions and IndieWeb-related development activity on GitHub.

However, it is worth noting that most IndieWeb-related GitHub repositories had a small number of people who contributed by either committing code or discussing issues. In fact, the median number of people who committed to IndieWeb-related repositories was 1, and repositories that were identified as strongly related to IndieWeb had fewer contributors than projects that were identified as only weakly IndieWeb-related. This indicates that actual development activities tend to focus on personal projects and small collaborations. This is further supported by noticing that as the number of people contributing to IndieWeb-related GitHub repositories has grown, so has the number of such repositories.

From this, it is evident that a relatively small number of individuals exhibit a strong influence

over discussions about how IndieWeb should be built. Since individuals at the centre of IndieWeb's chat network are more likely to be active on GitHub as well, this influence extends to processes of building IndieWeb software as well. However, insofar as most IndieWeb projects identified on GitHub are built by individuals and in small groups, decision-making about their development is characterized by decentralization. This is commensurate with my experiences building *Yarns* as reported in Chapter 4, where I performed much of the work of building on my own (albeit with influence from and collaboration with a small number of other IndieWeb developers), but my overall approach was structured by the Microsub specification and norms about modularity and interoperability.

The measures used in this chapter highlight centrality according to the frequency of one's contributions, and to a limited extent the ability of one's contributions to span the community (e.g. by measuring betweenness centrality). Where I have described a *centre* of influence, it is important to contrast this from the type of centre that emerges through hierarchical organizational structures or characterized by the centralized technical architectures of most corporate platforms. Specifically, none of the activities measured to identify this centre are restricted by hard barriers; there is no organizational hierarchy or membership process for either participating in IndieWeb's chat or building IndieWeb-related software.

The major limitation of this analysis is that is unable to address questions about what conditions may contribute to people's inclusion or exclusion from this centre (i.e. related to the second half of RQ3). I assert that questions of exclusion and inclusion are best addressed with a more intimate perspective than has been possible through the structural analysis I have used in this chapter. To that end, Chapter 6 investigates questions of inclusion and exclusion, relying in large part on results from interviews with IndieWeb contributors.

Chapter 6

Accounts of community from the ground: Investigating influence and exclusivity

This chapter continues the investigation of IndieWeb’s community structure that I started in the previous chapter. Chapter 5 began to address RQ3 through a description of structural features of IndieWeb’s community. As a reminder, RQ3 asks, *Is power over planning and executing design activities centralized or decentralized among stakeholders, and is anyone included or excluded as a result?* Chapter 5 identified that chat discussions are dominated by a strong core of very active participants. It mapped out general topics of discussion and identified how the frequency and longevity of chat participation as well as the topics one posted about were predictive of contributions to IndieWeb-related GitHub repositories. This has helped to identify a centre of influence but has not addressed questions of *who* may be included or excluded from this centre or *why*.

The question of exclusion is important in multiple dimensions. First, it is clear that IndieWeb generally follows a *self-management* model. In such communities, Downing (2001) asserts, “There is a standard danger that, although they may be highly democratic internally, they may become politically insulated from the ongoing push and pull of social life, a little self-sustaining oligarchy” (p. 70). Thus, identifying people who are particularly influential, as well as those who may be excluded from a community, contributes to an understanding of what viewpoints, goals, and experiences are present or absent from decision-making processes. Second, inclusion and exclusion are important for understanding IndieWeb’s contribution to “self-empowerment,” which I identified as one of its core values in Chapter 4. In that chapter, I agreed with Roedl, Bardzell, and Bardzell’s (2015) assertion that the potential for “empowerment” as presented in literature about maker culture can only be achieved with consideration toward barriers of access. This includes addressing if and how such barriers relate to existing forms of social injustice. Doing so gestures to a vision of empowerment that includes the disenfranchised in its pursuit of individual autonomy. A major theme I will present in this chapter is that IndieWeb’s contributors place a high value on efforts to welcome, teach, and support newcomers and outsiders, yet this has often been challenging. Ultimately, decisions related to exclusion and influence in IndieWeb’s community are articulations of IndieWeb’s values as much as any technical implementation, and are therefore

significant for understanding the processes of building and maintaining IndieWeb as a sociotechnical system.

To serve that line of inquiry, this chapter identifies factors that have contributed to inclusion and exclusion in IndieWeb’s community. Results are organized in three sections. First, I report demographic information about interview participants. While it is not possible to generalize these demographics to represent IndieWeb’s overall community, this helps to identify what set of views are included in the results. The second section presents results concerning influence and exclusion oriented around technical experience, and the third section discusses exclusion related to identity (focusing on a gender gap), as well as strategies for addressing such barriers. This distinction is based on categories articulated by interview participants, who generally spoke of technical and cultural obstacles as distinct from one another. However, this is not intended to accept a dualism between the technical and the social. Such dualisms have themselves contributed to the gendering of engineering work by mapping on to “masculine instrumentalism” and “feminine expressiveness” (W. Faulkner 2000). Instead, the two domains overlap, and this overlap is a significant theme in this chapter.

6.1 Overview of interview participants

Results presented in this chapter are informed by both interview responses and my participant-observation. Prior to reporting these results, this section will describe the interview participants. Chapters 4 and 7 also include responses from interviews, however interview participants’ backgrounds are especially relevant for this chapter. The description presented here is intended to help readers understand what perspectives are represented through the interviews, which is important for acknowledging the capacity of this chapter to address questions of diversity and inclusion in IndieWeb’s community.

Table 6.1 presents a summary of interview participants’ demographics. Rather than presenting a list of categories from which to choose, participants were asked to respond to demographic questions in their own words. In brief, interview participants were predominantly white, male, educated, and living in the United States. 80.0% (N=11) identified their race as “white” or “Caucasian.” 80% (N=12) were male. 80% (N=12) had earned a bachelor’s degree or higher, including 33% (N=3) with a master’s degree and 13% (N=2) with a doctorate. 73% lived in the United States, while the remaining participants lived in Europe. Given that the recruitment strategy for interviews was not random and the sample size is small, these figures are not generalizable to IndieWeb’s community more broadly. However, they do present a vignette of the type of individuals who are most active and prominent within IndieWeb’s community.

As described in Chapter 3, people asked to participate in an interview were identified based on their participation in IndieWeb’s chat channels and their contributions to notable IndieWeb projects. As a result, these participants skew disproportionately toward the core of active contributors to IndieWeb, including two of IndieWeb’s co-founders who have remained active in IndieWeb’s community. Table 6.2 presents a simple comparison of chat activity among interview participants and other chat users. Interview participants had a significantly higher median of messages posted to chat (overall and monthly), count of active chat months, and chat betweenness centrality. In sum, the interview participants included in this study are enthusiastic participants in IndieWeb. By definition, these individuals are not excluded from this community, so when they discuss exclusion they are usually identifying barriers that they have observed but not encountered personally, or that they have encountered but which have not been insurmountable.

Table 6.1: Demographic data about interview participants.

Identified race as	N	%
Black/Latinx	1	6.7%
Jewish	1	6.7%
Turkish-American	1	6.7%
White	11	73.3%
White/Hispanic	1	6.7%

Identified gender as	N	%
Female	3	20.0%
Male	12	80.0%

Highest level of education completed	N	%
High School	3	20.0%
Bachelors	5	33.3%
Master's Degree	5	33.3%
Doctorate	2	13.3%

Country of residence	N	%
Austria	1	6.7%
Germany	1	6.7%
Italy	1	6.7%
Sweden	1	6.7%
United States	11	73.3%

Age	N	%
20-29	4	26.7%
30-39	4	26.7%
40-49	5	33.3%
50-59	1	6.7%
60-69	1	6.7%

Table 6.2: Comparison of chat activity among interview participants and other chat users.

	Interview participants	Other chat users
Median total messages	8776	5
Median messages per month	333	4
Median active chat months	41	1
Median chat betweenness	0.1432	0.000

6.2 Influence and exclusion around technical experience

6.2.1 Influence through technical implementation

During interviews, I asked participants if they felt they had been influential in the IndieWeb as well as who had influenced them. Technical expertise was commonly cited as a factor when identifying influence from others. Among interview participants who were deeply engaged in IndieWeb development, technical influence was most commonly identified based on seeing and being inspired by what others are working on or how they work.

It’s inspiring to see how he is able to create projects that are handling a large volume of data and providing a valuable service, and yet [...] it’s still cheap enough for him to run and it’s also not something he has to handle too much active support on. [...] I try to keep that in mind as I’m building things out. (Interview participant #15)

Another form of influence is actively sharing knowledge with others. This was observed most commonly in interviews with individuals who had come to IndieWeb more recently and in blog posts written by people about their first encounters with IndieWeb.

I recently switched everything over to WordPress. One of the reasons for that was [a specific IndieWeb contributor] [...] is just super “yes you can do this with WordPress”, and “here’s how you do that”, and “here all the things are already finished so you don’t actually have to code any of the stuff yourself.” (Interview participant #5)

Both of these demonstrate the sharing of knowledge related to technical work. In general, community members whose technical accomplishments were referenced as influential were also likely to be recognized as actively teaching others and answering technical questions posted to chat or at in-person events. Additionally, some participants downplayed their own influence by contrasting themselves with others who had built and shared IndieWeb code or software:

I might have been able to steer some discussions toward certain points because I do like to participate in the standards discussions and so on. But in the end, I think the people who have the most influence are the people who are then actually building the thing and sharing all their code. (Interview participant #2)

Implementations were recognized as influential for resolving disagreements and conflict. Research about conflict has conventionally distinguished between task conflicts, which arise from “differences of

opinion about how a specific task should be performed,” and affective conflicts, which are “emotional and focuses on interpersonal incompatibilities” (van Wendel de Joode 2004, p. 105). In open source communities, affective conflicts have generally been conceptualized as being negative for team function, whereas task conflicts can contribute to diversity of opinions and have a positive effect (Filippova and Cho 2015). Interview respondents reported that most disagreements they had were related to technical decisions and did not develop into affective conflicts. Technical disagreements were described as being addressed through implementation, which is argued to identify solutions more effectively than arguing:

If I have an idea for a feature that I think should work one way and somebody else thinks it should work a different way, then there’s no sense in arguing or getting mad at each other because that’s not a good way to make progress.

So instead I try to actually build it out [...] because either the thing that I build will be perfectly functional and the other person will say, “Oh, okay, that’s great. That’s working now.” Or I will build it out and be like, “Oh yeah, it turns out that was actually a really awkward design decision and it shouldn’t work that way.” (Interview participant #15)

At the level of inter-project conflict, IndieWeb’s commitment to plurality was reported to diffuse task disagreements by allowing multiple solutions to co-exist.

Because it’s not a monoculture and there’s lots of different projects one project can use one approach, another project can think about things a little bit differently, and that’s actually how you get to a really robust distributed decentralized community.

(Interview participant #14)

This is explicitly referenced in IndieWeb’s wiki, which contrasts IndieWeb’s approach to open source *monocultures* in which everyone works on the same software or system:

Open source monocultures can foster unproductive disagreement between collaborators. If everyone is working on the same project then we all have to agree on all details of implementation. Alternatively, focusing on interoperability and personal use cases encourages healthy debate without the need for anyone to compromise. (IndieWeb.org 2019f).

As reported in Chapter 4, a caveat to plurality is that early implementations establish a foundation of features and approaches that influences the design of subsequent IndieWeb sites and software, especially those that interoperate with the earlier implementation. This was articulated by several interview participants who jokingly referred to a “single point of Aaron” among IndieWeb’s building blocks, a phenomenon described on IndieWeb’s wiki:

single point of aaron [sic] is when a relatively early IndieWeb technology is being developed and there haven’t been enough implementations to create proper diversity causing an over reliance on one of Aaron Parecki’s services in order for the technology to work broadly. The goal is always to mitigate this by encouraging people to build more implementations of a given technology or to discover what might be causing a lack of implementations.

Early implementations contribute to infrastructural momentum (Star 1999), establishing conventions that become imbricated in processes of interaction among IndieWeb building blocks. Further, as will be further discussed in Sections 6.2.2 and 6.2.3, there is a divide between the possibilities for decision-making among developers and other IndieWeb users. People who use IndieWeb software created by others, instead of building software for themselves, inherit decisions made elsewhere.

6.2.2 Formal technical barriers

In IndieWeb’s early period, there were explicit technical barriers for participation. As described in Chapter 4, participants at the first few IndieWeb events had to demonstrate they were builders/creators¹—people who produce code, design, or UX resources related to IndieWeb—in order to attend.

During a chat discussion in 2013, Çelik described that restrictions about who could attend IndieWeb events deliberately established a hierarchy of IndieWeb contributors:

either you’re a serious indieweb/socialweb creator - and are proving it with your own site

or you’re not serious, and you’re a peripheral contributor

[...]

that kind of “class” distinction is explicit and new about indiewebcamp

since it’s not by who-knows-who

but by what-have-you-done

[...]

and it goes against the assumed egalitarian nature of all things internet

since we’re explicitly stating there are two tiers

creators and everybody else

(Excerpt of messages posted to <https://chat.indieweb.org/2013-04-01>)

The term “class” here should be interpreted as a misnomer—When I spoke to Çelik about this statement, he clarified that he was really referring to an “engagement” distinction based on how one puts IndieWeb’s philosophies into action. Furthermore, the move away from “who-knows-who” and toward “what-have-you-done” is a refutation of conventional class hierarchies. With that caveat, I include this quotation to highlight how IndieWeb was initially structured with these two tiers in mind.

One way that the restriction to builders only was enforced by a technical requirement for RSVPing to IndieWeb events. In order to RSVP, one had to have implemented some IndieWeb features on their site. One interview participant described frustration at this requirement.

When IndieWeb first started [...] to go to an IndieWebCamp they wanted [you] to authenticate an RSVP with your own personal website. To do that you had to set up some things on your personal website to make that work[...] You had to be enough indie to be able to RSVP from [your] website to go to IndieWebCamp. and that’s just bogus. So, a few of us tried to push back on that concept. That’s a great way to immediately ensure that a bunch of people never even find out what your event is about [...] if they immediately feel excluded and isolated because they can’t even figure out how to say like, “Sure I’m interested in going to this.”

(Interview participant #11)

This requirement was not strictly enforced, however, as the interviewee explained they attended events without RSVPing:

¹The term “builder” in an IndieWeb context has been used interchangeably with “creator.” As a result, quotations presented in this section use both terms as equivalent.

I would always get frustrated and say, well, I'm just not RSVPing because I can't figure out how to make this work. So maybe I'll come. Maybe I won't go. But I'm not going to tell you if I'm going to come or not. (Interview participant #11)

Further, from the first IndieWebCamp in 2011, there was an option to attend as an “apprentice” to a “builder,” who would RSVP on their behalf and be “completely responsible for them” (IndieWeb.org 2019a).

This requirement was eventually eased and then removed altogether. Alternate RSVP mechanisms were provided for some but not all IndieWebCamps as early as 2012. By IndieWebCamp 2014, the term “builders” has been recast as “creators,” and a new class of attendee was encouraged, bloggers:

If you're an active blogger (i.e. at least once a month) yet don't feel like you would call yourself a creator, it would still be great to have you participate in IndieWebCamp. (https://indieweb.org/2014/Guest_List, Accessed Jan 31 2020)

By 2015, anyone who wanted a personal website or to contribute to IndieWeb was encouraged to attend as a full participant (not an apprentice). Additionally, when this barrier was active, it did not apply to participating in IndieWeb's chat channels, which could be accessed by anyone who could access IRC (eventually IndieWeb's IRC channels were bridged to also be accessible via Slack, Matrix, or a Web browser).

The lowering of those barriers is not a move away from distinction between “creators and everybody else” but instead relates to IndieWeb's technical foundation being established. Early in IndieWeb's history, it was difficult for people who lacked Web development experience to engage in the activities of an IndieWeb “creator.” Thus, even though IndieWeb was never developers-only, it was developer-centric. Hard barriers to entry were lowered early in IndieWeb's history, both through changes to event policies and because, by 2014, there were a variety of existing tools available that people could use to implement IndieWeb features onto a personal website without having to understand their inner workings. As such, the technical bar to be an IndieWeb “creator” was lowered. However, even as IndieWeb has become easier to use, it's developer-centric origins cast a long shadow of informal technical barriers, which I will discuss presently.

6.2.3 Informal technical barriers

Although formal technical barriers for attending IndieWeb events have been removed, IndieWeb is still characterized by a relatively high level of technical complexity, creating challenges for growth that have been a regular subject of discussion. In 2014, amid a period of growth in the number of individuals participating in IndieWeb's chat (see Figure 5.2) and IndieWeb's network of GitHub repositories and contributors (see Figures 5.12 and 5.13), a model of IndieWeb “generations” was developed to describe a trajectory for increased adoption (IndieWeb.org 2020b). This model describes an incremental adoption of IndieWeb by four generations, defined by their degree of experience with Web development:

- Generation 1: *Development leaders* who are skilled in Web development, capable of building their own software and running a Web server.
- Generation 2: *Journalists and bloggers* who have a domain name and a blog, and understand the basics of managing a website using software such as cPanel, Softaculous, or Fantastico.

- Generation 3: *Personal domains managed by third parties* consists of people who have a domain name pointing to content hosted by third parties such as WordPress.com or Tumblr.
- Generation 4: *People using social networks* who have never purchased a domain name or run a personal website.

Ironically, the generations model itself has been observed as a potential barrier for growth. The ranking implied by this model has been criticized by members of IndieWeb’s community, such as in chat messages by Parecki in 2019:

The problem with generations is it’s an ordering, and in reality there isn’t an order and people don’t try to move “up” the ordering, and often people exist in multiple of the categories at the same time. [...] I’ve seen people look at the generations page and then talk about how they don’t feel like they are high enough in the ranking and that just isn’t something we want

(Excerpt of chat messages posted to <https://chat.indieweb.org/dev/2019-03-02#t1551486579345700>)

In the same discussion, it was asserted that the intention of the generations model was not for individuals to self-identify, but rather to describe a phenomenon that was underway:

I believe [generations] was intended to be a fairly literal time-based metaphor, not one on ranking or any sort of judgment [...] IndieWebCamp did literally start with “development leaders”, and grew quickly within a few years to incorporate “journalists and bloggers” which is when this diagram was made, 2014

(Excerpt of chat messages posted to <https://chat.indieweb.org/meta/2019-03-02#t1551492667791200>)

From the latter perspective, the generations model has generally predicted an accurate timeline of IndieWeb adoption. Most of IndieWeb’s earliest community members were squarely part of generation 1. By 2014, some of these “development leaders” had built tools that cater to generation 2, such as plugins for self-hosted instances of WordPress² and other content management systems, and hosted services for receiving Webmentions (e.g. <https://webmention.io>) and following IndieWeb’s POSSE syndication flow (e.g. <https://brid.gy>). In September of 2014, an IndieWeb-compatible CMS called *Known* was launched (IndieWeb.org 2020d), which offered a hosted service similar in complexity to WordPress.com or Tumblr, and a similarly easy-to-access IndieWeb platform called Micro.blog was launched in 2017 (IndieWeb.org 2020f).

However, as IndieWeb’s community of users has diversified, newcomers’ experiences have been inconsistent. The creator of Micro.blog expressed that Micro.blog users are sometimes overwhelmed once they start learning about IndieWeb more generally, asserting that IndieWeb’s technical complexity is related to its the technically skilled users that dominated its early days:

Sometimes I see those users, they use Micro.blog and then they discover the IndieWeb [...] and they get really lost and confused and they give up. I know that the IndieWeb is aware of this, but I do think it’s gonna be really important for it to become more accessible to

²Note: *self-hosted instances of WordPress* refers to cases where an individual has installed WordPress to their own server. This is distinct from using creating an account with *WordPress.com*, which provides access to pre-configured installation with no need to manage one’s own Web back-end.

mainstream users because you shouldn't need to know how to program to be a member of the IndieWeb community. And no one says you should be. But because it started with so many technical users and programmers, it can sometimes feel that way.³

(Interview with Manton Reece)

A clear influence upon IndieWeb's technical character is its building block approach, where most IndieWeb sites and software are built by adopting and combining features individually. Chapter 4 has argued that this approach generally supports developers' autonomy because they can choose which features to implement, and individual implementations can help shape the definition of standards and other coordination mechanisms. In fact, *developing* for IndieWeb is generally simpler than for many other decentralized Web projects; one can pick and choose a small set of features to learn about independently from each other, and foundational building blocks such as MF2 use simple HTML and CSS conventions that are easy to understand for almost all Web developers. However, from the perspective of people who want to use, but not develop, IndieWeb software, the building block approach can be a source of confusion. Owing to the fact that most IndieWeb sites use a combination of many moving parts, keeping things working smoothly has sometimes required keeping up to date on others' development, even for people who operate an IndieWeb site but generally rely on others' software rather than writing code themselves. This was described as especially relevant in earlier days where IndieWeb building blocks were in earlier stages of development:

For awhile, I've almost felt like you had to keep up in [chat] or follow development on GitHub [...] I remember those days where it was like, yeah, you have to pay attention to what's going on and what the updates are to understand.

(Interview participant #13)

In many cases, IndieWeb's building blocks and the software that implement them have matured so that they can usually run smoothly in the background. By contrast, emerging functions are still subject to change since they are often in the process of having bugs identified and resolved through use. As the analysis of IndieWeb's chat and GitHub archives has shown, a large proportion of IndieWeb's discussions are dominated by an extremely active core of individuals, who are likely to have been active in IndieWeb's community for an extended time. Moreover, that analysis has identified that individuals who post a large number of messages to chat were more likely to be active on GitHub, as were individuals whose chat messages were related to the "events" topic. This suggests that people first encountering IndieWeb either at an event or in chat will be exposed to conversations and presentations of recent technical developments. Certainly, this matched my experience, especially of attending in-person events, where much of the focus of discussion was on future developments. This could result in newcomers wanting to use brand-new IndieWeb features or software that were in early stages of development and not (yet) intended for anyone but developers:

We get it all the time where people are like, "Oh I would love to do the stuff that you guys show at a demo." Like, okay yes, but the thing we just showed as a demo took three Web developers a month to put on and there's no one-click installation. That barrier definitely exists.

(Interview participant #2)

This example and the others noted above demonstrate that newcomers to the IndieWeb may quickly encounter technical features that are yet to be polished. IndieWeb's community spaces simultaneously

³In an email correspondence later, he qualified this statement by explaining that he is a big supporter of the IndieWeb and sees its potential to reach regular users (Reece 2020).

cater to seasoned developers who want to extend its features at the cutting-edge, and newcomers who want to set up their first website. This contributes an implicit hierarchy.

6.3 Influence and exclusion around cultural diversity

The previous sections have identified exclusion and influence based on technical experiences. This section discusses forms of exclusion related to gender, race, ability, and other categories of marginalization, as well as strategies for addressing these barriers. Whereas technical barriers for becoming a central participant in IndieWeb’s community are rooted in deliberate decisions during its origin, the challenges discussed in this section are related to historical exclusions far beyond the scope of IndieWeb itself. Even in communities that emphasize outreach to excluded groups, broader patterns of exclusion from society can lead some to feel they don’t ‘fit in’ (Taylor, Hurley, and P. Connolly 2016). Gender has been an especially prominent barrier among maker and hacker groups, which, like tech in general, are dominated by men (MacArthur, Wong, and Hancock 2019; Whelan 2018). IndieWeb is no exception, and is visibly white and male. This has been a frequent subject of discussion among IndieWeb’s organizers and community members, with much of the emphasis on the community’s gender divide. During interviews, gender was the most discussed factor for identity-based exclusion, and this section will focus on IndieWeb’s gender gap.

6.3.1 Gender gap

The proportion of women to men has been a consideration from the first IndieWebCamp event in 2011. There were 36 attendees, 75% men and 25% women. After the event, a revision was made to the 2011 guest list article in IndieWeb’s wiki commenting on the final count:

From the builders + apprentices it appears the final IndieWebCamp Classof2011 count is:

- 36 total (9 women = 25%)

Compared to last year’s photo of the 2010 Federated Social Web Summit which had:

- 37 total (1 woman = 2.7%)

I’d say that’s progress.

(IndieWeb.org 2019a)

That balance of event attendees by gender has not remained steady. Some interview participants were frank about failures to maintain gender diversity:

If you’ve got a two-day IndieWebCamp or maybe a two and a half day thing, people might come or the first morning and then I would notice like, “Okay so Saturday morning there may be four or five women here and then Sunday afternoon I’m the only one who’s left.”

(Interview participant #11)

At Austin this weekend one person who attended commented that she was happy she wasn’t the only woman in the room. For half the day, because the other woman left. And you know our goal as a community is that nobody should feel they have to say that but, but that was what happened.

(Interview participant #12)

This highlights two dimensions to the gender gap. First, women have been less likely than men to participate in the IndieWeb in the first place, and women have also been more likely to leave events early. Most IndieWeb events are described on IndieWeb’s wiki along with group photos of attendees, sometimes showing who was present on both days of two-day events. These are not a complete record (many events only have photos for one day), but have been reviewed to corroborate reports from interviews. These photos make it clear that overall, attendance at IndieWeb events has been and continues to be skewed toward men. The phenomenon where women were more likely than men to attend only the first day of IndieWebCamps is evident in photos of some events, but not all. In both dimensions, there is evidence that the ratio of women to men at IndieWeb events is improving, though remains significantly less than fifty-fifty.

One of the most common suggestions from interview participants to improve diversity was to invite people from a variety of backgrounds. This was identified as a challenge because the personal networks of white men working in tech are often themselves not diverse:

Sometimes at the events it looks like there are only white dudes at an event, which we’re trying to actively fix. But it’s always hard because you have to do the outreach and not everyone has a network that’s even interested sometimes. Like, they just aren’t there, or your network is your work. But if you’re working in Web development, chances are that everyone at your office looks like you anyway. (Interview participant #2)

This illustrates homophily (McPherson, Smith-Lovin, and Cook 2001) in two ways: First, in people’s professional networks, and second, the kind of people who share an interest in a particular community’s way of doing things are likely to share other aspects of life experience with members of that community.

Others suggested moving beyond personal invitations to actively engage with communities that are similarly oriented around technical engagement but have a stronger focus on gender and cultural diversity. However, advocates of this approach acknowledged that IndieWeb has struggled to communicate its value, sometimes presenting its technical building blocks as though they speak for themselves:

I kind of wish we did more outreach toward [other people-centric] projects. Talk to them. [...] To come to show what we have to offer and [how] we could integrate to them. I feel like sometimes when we come to other projects, we’re like, “Hey look,” [thumps table] “Microformats.” [laughs]. It doesn’t necessarily work with everybody all the time. (Interview with Jacky Alciné)

To encourage more diversity in the IndieWeb, one interview participant asserted, requires IndieWeb’s contributors to more actively participate in other communities as well.

You have to participate in their communities as well. [...] Helping out with those communities as well and really more deeply embedding in what those communities are doing. By doing so you get a much deeper sense of what is actually important to this more diverse set of people. [...] It’s got to be about, okay, what’s important to them? How can we meet them where they’re at and be useful in their lives? (Interview participant #14)

In sum, outreach was identified as an important step for addressing gender and other diversity gaps. Furthermore, while cultural homogeneity in people’s personal networks poses a challenge for diverse

outreach, participating with other communities was identified as a way to broaden one’s network and to understand the needs of others.

However, In the examples noted above, women who attended IndieWeb events have historically been more likely than men to leave early. Outreach and invitations are unlikely to be sufficient for solving this problem. One factor is that IndieWeb events can occupy time that is disproportionately available to men and women. Past research indicates that women take on a greater share of unpaid social reproductive labour, such as childcare and housework, while balancing paid employment (L. C. Sayer 2005) and that women feel more time pressure than men (Mattingly and Liana C. Sayer 2006). Accordingly, women may find it difficult to devote time to events held during evenings and weekends, as has been observed by Neff (2012) in her study of *venture labor* in Silicon Valley.

Another possibility is that behaviour or discourse has articulated gender biases. The first step to address this possibility to is consider whether women were subject to discriminatory behaviour or stereotype threat, the risk of being judged by negative stereotypes that undermine one’s performance or aspirations (Spencer, Logel, and P. G. Davies 2016). No evidence was observed to suggest this was the case. One of the women I interviewed explained that IndieWeb’s community is more welcoming than other technical environments she has experienced:

Despite the fact that I’m one of very few females involved in the IndieWeb community, nobody considers my gender when talking to me. [...] [At work] I turned up at a meeting and I’m there taking notes, and everyone’s like, “Oh yes, secretary taking notes” and then my boss is like, “So this is the person that does all of the actual [software development] work.” And you can see the mouths go, “Oh.” That doesn’t happen here and that’s one of the things that I do like about it.

[...]

I think it’s mostly just a community mindset. Like nobody does it, so when new people join the community, they also don’t do it because they don’t see it happening. So, it’s fostering the right environment in the first place. (Interview participant #5)

In the absence of explicit stereotyping, there is also a possibility that a community’s discursive conventions contribute to exclusion. Reagle (2013) argued that “the anarchic–libertarian ethic” that is common in open source communities “requires a significant tolerance for adversariality that may be alienating to some participants.” This troubles the dichotomy between “task” and “affective” conflict that was introduced in Section 6.2.1. Specifically, it highlights that task conflict can quickly become affective, and so drawing too firm a distinction between these modes of argument can reinforce the dualism between “masculine instrumentalism” and “feminine expressiveness” (W. Faulkner 2000) that I referenced in this chapter’s introduction. I observed some evidence of adversariality in IndieWeb chat where, upon raising a criticism, one was likely to be asked to justify it with real-world examples. This was commented upon by an interview participant:

If you want to bring up a criticism then you’ll often be asked, “Well where’s your citation?” [...] That’s a heteronormative argumentative pattern. It’s basically like, “Prove it, prove it, prove it, prove it.” That’s a male discourse pattern. (Interview participant #8)

Although the trend of demanding real-world examples is an example of adversariality that could be discouraging, it is also related to IndieWeb’s overall technical strategy of building for one’s own needs. As

reported in Chapter 4, this has been presented as a means for efficient progress by building technologies whose utility is demonstrable rather than relying on hypotheticals. However, this strategy of focusing on problems that have been observed and documented within the community can lend momentum to biases. There has been disagreement about the long-term effects of this model for diversity:

One of the biggest disagreements I have with the IndieWeb community is this idea of scratching your own itch [...] if your community is biased towards a certain demographic, it actually just perpetuates that because all you're doing is you're building to that demographic.

(Interview participant #14)

When asked about potential barriers related to gender, race, culture or other factors, it was common for interview participants to acknowledge their blind spots:

I often have a hard time seeing [those barriers] which is the obvious, just another white tech dude in the same community sort of problem. Very hard for me to talk about those issues.

(Interview participant #2)

There is some nuance here. When we discussed this comment after our interview, participant #2 expressed some reservation because he identifies as male but sometimes uses gender-neutral pronouns and has, in fact, considered gender carefully in relation to his own experience. Nonetheless, he agreed with my general conclusion that building for oneself could exacerbate obstacles for inclusion, since “white tech dude” developers lack knowledge about the experiences of others. A caveat to that critique is that interview participants overwhelmingly acknowledged their privilege and were eager to better understand how to build for others. For example, several interview participants explained that they had accessed their own websites using screen readers for people with visual impairment, so as to better understand how they could improve accessibility. Furthermore, IndieWeb has generated several applications, services, and platforms that go far beyond “scratching their developers’ own itches.” These tend to be the most widely used and impactful IndieWeb developments. *Known* and *Micro.blog*, for example, have been designed to cater to communities first and foremost, with design decisions based on other people’s needs more than those of their developers. During interviews, another developer began to argue that “the IndieWeb projects I’ve done to scratch my own itch have been more successful than the ones that haven’t” before reflecting on a number of cases where he had built tools that he did not personally use and concluding, “a lot of stuff I’ve done might be used even if I haven’t” (Interview participant #4). Thus, while IndieWeb’s principle of building for one’s own needs has ramifications for inclusion, it is not generally interpreted as a restriction (i.e. that one should not build for others).

Finally, the technical divide described earlier in this chapter is itself imbricated with gender biases. S. Faulkner and McClard (2014) assert that women are more likely than men to come to making communities through a “non-technical path” from an “art, craft or design background” and “more interested in what technology enables, rather than in the technology itself” (p. 187). According to interview participant #3, the early decision that IndieWeb was for “builders only” was carefully considered as means to challenge the “default tech dude culture”: “Rather than just saying ‘we can’t have too many tech dudes,’ we tried to identify the types of attitudes and behaviours we wanted to avoid” (Interview participant #3). One of those behaviours was the tendency of “blowhards” to talk about technology rather than building with it, who, participant #3 reports, are almost always men. Participant #3 went on to explain that “building” was to include a range of activities to suit different backgrounds, including coding, graphic and layout design, and user experience design.

We recognized that it was not enough to discourage the problem folks we'd encountered, but that we needed to complement that with explicitly encouraging the folks that were usually put-off by [coding and technical environments]. (Interview participant #3)

This effort to challenge barriers to inclusion has been evident from the start. However, the reality is that the majority of IndieWeb's most active contributors are developers, and conversations and activities among these contributors tend to emphasize technical details in ways that can discourage people who come to the community through a "non-technical path" as evidenced by another interview participant:

I have pointed out how this very dude-heavy IndieWeb movement is not going to attract a diverse population if they don't just make it relevant to people who probably have other interesting things they could bring to the mix that are just not building plugins or whatever. Developers is only one piece of the pie. We have never really argued that exactly because they're well aware about these things and I think they would like to change them, but that is also the downside of a really independent movement. That it's also nobody is in charge. Nobody says ok we're making this happen and everybody get on the same page.

(Interview participant #7)

In general, these observations demonstrate a community that strives to treat everyone as equal, but which has inherited gender and other biases that are common to open source and other technical communities more generally. As suggested through this section and explicated by this interview participant, individually-focused interventions are insufficient for resolving these problems, which originate from society-wide systemic issues.

6.3.2 Addressing exclusion at a community level

The previous section identified obstacles to IndieWeb's cultural diversity, focusing on gender imbalance. Many of these obstacles are reproductions of broader historical patterns in which communities oriented around technology are predominantly white and male. In a review of research about organizational diversity, Shore et al. (2009) identified that diversity was most often achieved through action at group and organizational level, and that a philosophical commitment to diversity among management was particularly important. With regard to looser communal structures such as IndieWeb's, Richard's *Inclusive Communities of Practice Framework* identifies that efforts to promote underrepresented cultural groups at a community level have worked to address individual factors, such as those related to social group identification and other sociocultural factors (Richard and Gray 2018). Action at a community level is likely to transcend what is possible through individual action when it comes to addressing diversity and inclusion. This section reports on efforts to address diversity at IndieWeb's community level, especially facilitated by community organizers.

Although IndieWeb does not have a conventional hierarchy, a number of members are recognized as "organizers." The pathway to be recognized as an organizer is to participate in event planning. Organizers are "everyone who has co-organized an IndieWebCamp or organized Homebrew Website Club meetups in the past two years" (IndieWeb.org 2019g). Additionally, some participants can act as informal community leaders, typically by virtue of longstanding or active participation:

We try to act as community managers keeping the community healthy and positive. And trying to, by our very actions in the community, exhibit principles and the code of conduct

in particular, to keep it welcoming. That’s probably the biggest, most important role that a bunch of us play in the community. (Interview participant #3)

The most visible tool for addressing exclusion based on identity is IndieWeb’s code of conduct, which was first drafted in 2013, and at the time of this writing is summarized as follows:

Be respectful of other people, respectfully ask people to stop if you are bothered, and if you can’t resolve an issue, contact staff. If you are being a problem, it will be apparent and you’ll be asked to leave. (IndieWeb.org 2019d)

“Staff” in this case refers to either the organizer(s) of an event or designated community organizers. In June 2018, a statement was added to the code of conduct about resolving conflicts in chat:

If you believe you’re experiencing practices in chat or at an event which don’t meet the policies above, please immediately contact any ally you may know. Anyone in the community should feel empowered to resolve issues and uphold our high standards.

However if you feel the issue requires escalation to a community organizer please contact: [list of designated contacts] (IndieWeb.org 2019d)

This addition was decided upon at an organizers meeting in which one individual explained that they had been contacted to resolve an issue that occurred in chat “because [the person seeking resolution] see[s] me around a lot, and had seen I was online at the time of the incident” (IndieWeb.org 2019b). In this case, the individual identified as a de facto leader was also a formally recognized “organizer,” but it was their regular presence in chat that denoted their suitability to mediate a conflict. This highlights two features of how leadership is defined. First, that “anyone in the community” can step up to moderate conflict, and second that being active and visible in chat has been interpreted to indicate that one is a de facto leader, or at least versed enough in the community to navigate conflict.

Organizers meet at least yearly. In-person meetings are held before IndieWeb’s annual summit in Portland, OR, USA, and smaller organizers meetings are held before some IndieWebCamp events. In both cases, people who are not physically present may attend remotely. Summaries of issues discussed at organizers meetings are published on IndieWeb’s wiki (see “Issues” at IndieWeb.org 2019g). Although organizers meetings are only open to recognized “organizers,” anyone is invited to add an issue to the wiki, which will then be addressed at the next meeting. However, I did not observe any instances of issues being added by people other than organizers. While issues of inclusion have been regularly discussed in chat and at events, organizers meetings provide possibilities to address issues systematically:

The folks that have really stepped up as organizers are pretty firmly in alignment on improving the demographics and the [diversity and inclusion] as a top priority. Like something that we talk about every time we have an organizers’ meetup or organizers’ summit, one of our top topics is [...] how are we doing in terms of diversity, making it a more inclusive, safer space, and what can we do better? (Interview participant #3)

I will briefly identify two strategies that have been subjects at organizers meetings. First, there has been an effort to increase diversity across all IndieWeb events.

We try to encourage any IndieWebCamp or Summit to try to have someone who does not identify as male, try to have someone who does not identify as white, if you can. Ideally both. Because it makes a massive difference in terms of organizing, in terms of prioritizing,

in terms of outreach, and just in terms of when people look at a conference like, “Who’s running the conference?” [...] [At IndieWeb Summit 2019] we have five keynote speakers. One woman, one person of colour, out of the five. And that’s a big step up for us too. I think last year we had just one woman. So, we would like that to be preferably at least 50 per cent women, but we’re not there yet. We looked at the speakers that we’ve invited for the IndieWebCamps over the past year, and IndieWeb Summits, and we are about at 40 per cent of the invited speakers were women over the past year. (Interview participant #3)

Another intervention at that organizers level was the creation of a travel fund for underrepresented people. The first attempt to distribute the travel fund was unsuccessful:

We had someone who submitted an application and then by the time we saw it, it was too late to actually go book the ticket and they had made other plans for the date. We just didn’t have a good process in place for actually handling those submissions. [...] That was something we had actually talked about last year [at an organizers meeting in] Berlin to try to again formalize that process. (Interview participant #15)

This resulted in an improved process, including a designated organizer to manage the travel fund for IndieWeb Summit 2019.

This year we’ve got started earlier, we approved a whole bunch of folks earlier, and now I don’t know what the last count is but it’s at least three maybe four people. And that’s a start. I think that’s somewhere between 5-10 per cent of the attendees which is pretty good. We’d like it to be higher but it’s a start and we learned a lot from the process this time and we know how to make some things simpler for next time. (Interview participant #3)

The early problems with the travel fund demonstrate a pitfall of an informally-defined leadership structure. Forms of outreach that include funding or complex organization across multiple events require coordination and formalized processes. Overall, IndieWeb has much room for improvement when it comes to its diversity, but the move toward institutionalized inclusivity strategies demonstrates a meaningful step.

6.4 Discussion and conclusion

In this chapter, I have addressed the second part of RQ3 by identifying factors contributing to exclusion in IndieWeb’s community. Where the previous chapter identified a divide between a core of extremely active IndieWeb participants and a large periphery, this chapter investigated what is involved in being part of the core, identifying potential forces of exclusion in that process. It is important to note that no strict or hard barriers to entry were identified. IndieWeb has no strict gatekeepers nor criteria for membership (or a concept of *membership* at all). Yet, a system that is decentralized and open such that anyone can participate develops centres of power based on which individuals have the greatest access to the practices that define participation.

I have identified IndieWeb’s emphasis on technical aptitude and eagerness as contributing to an implicit hierarchy. Early IndieWeb events were focused on attracting *builders only*, and much of IndieWeb’s early emphasis was on the definition and development of the technologies that would facilitate its subsequent growth. This was reasonably regarded as a pragmatic move, though IndieWeb’s early emphasis on

developers has been regarded as contributing to challenges for newcomers with different types or levels of technical experience. Moreover, building and sharing IndieWeb software was regarded by most interview participants as the central way of influencing IndieWeb's community, even this chapter and Chapter 4 have identified that coordinating events, establishing and upholding community policies, documenting best practices, and related activities are vital and influential.

It is worthwhile here to refer back the concept of communities of practice. IndieWeb's community can be characterized as a community of practice in that its participants are engaged in a shared set of practices of using their personal websites as their primary online identity and building tools for enabling this. Consensus in a community of practice is defined by *mutual engagement* through common activities, *joint enterprise* toward a common goal, and a *shared repertoire* of discourses and meanings (Wenger 1998). IndieWeb's technical infrastructure, documentation, and community technical support have matured to a degree that a wide variety of people can mutually engage in using their personal website in a way that follows IndieWeb's conventions. This is part of a joint enterprise of creating a social network of personal websites, and in the process owning and controlling one's data and online experience. And in the process of engaging in this process, participants can develop a shared repertoire of meanings, for example concerning websites, standards, and software around which IndieWeb's community is organized. And yet, developers have privileged access to an additional set of practices around writing code, commenting on others' code, and actively contributing to design decisions. These are precisely the practices that were observed in Chapter 5, and thus the technical aptitude described in this chapter is a valuable trait for becoming an influential participant in IndieWeb's online community.

Further, centres of power derived from technical acumen inherit historical biases related to privilege. Although I have focused on privilege related to gender in this chapter, this pattern extends also to other forms of privilege, most notably related to race. Rather than resulting from decisions within IndieWeb's community, exclusions based on privilege derive from longstanding gendering and racialization of technical practice in broader society.

Attempts to improve cultural diversity have echoed a pattern identified in Chapter 4. That chapter showed individual autonomy was among the most central values motivating the design of IndieWeb's technical architecture, yet certain types of challenges were only met when served by more communal values such as consensus and conformity. As for actions taken to improve inclusion, it is clear that both technical and other forms of exclusion are related to individualism. Technical exclusion has been related to patterns of highly adept developers building for their own needs, and in the process creating software that has at times been confusing for people with more limited technical experience. And the potential that individuals can address inclusivity problems through personal invitations is constrained by their personal networks and their life experience. As has been observed in similarly oriented communities, the notion that individuals can pursue egalitarianism through technical participation reflects a universalism that "all but depends on glossing over differences in power, access, and status among different groups" (Dunbar-Hester 2014, p. 84). I have identified diversity initiatives coordinated at the community level as well-suited to addressing these challenges. Specifically, I have cited efforts to achieve diversity among keynote speakers and IndieWeb's travel fund. Insofar as IndieWeb's organizers are described as being "firmly in alignment" on diversity and inclusion, this type of coordination is unlikely to have necessitated compromising their autonomy. Nonetheless, it demonstrates the employment of communal values to address a situation in which IndieWeb's typical values of individual autonomy were insufficient.

In sum, I have used this chapter to identify and discuss factors that shape individuals' capacity to

become influential contributors to IndieWeb's community. I have focused on tacit barriers related to technical experience and structures of privilege, which intersect with one another. And I have argued that addressing these problems requires coordination and the employment of communal values. This has, of course, not been an exhaustive account of potential barriers, but has highlighted structures that affect access to shaping IndieWeb's values, architecture, and their relation to one another.

Chapter 7

Building Alongside the Corporate Web

7.1 Introduction

Thus far, this dissertation has concentrated on activities and relationships internal to IndieWeb’s community. Chapter 4 focused on an individual perspective of building for the IndieWeb. Through describing my own experiences building IndieWeb software alongside codified and shared descriptions of IndieWeb building, I found places where individual autonomy was influenced by community structures and norms. This led to my analyses in Chapters 5 and 6, which described centres of influence and factors contributing to exclusion among IndieWeb’s community.

The present chapter extends upon those analyses to consider an important relationship external to IndieWeb itself. Specifically, it investigates frictions and compromises that arise when IndieWeb operates alongside and even within the infrastructures to which it claims to provide an alternative. Although IndieWeb’s premise is that people use their own websites as their main online presence, they do not avoid corporate social media altogether. IndieWeb users usually syndicate content from their personal Web sites to Facebook, Twitter, and other *silos* (i.e. corporate platforms), and then *backfeed* comments, likes, and other responses back to the original post. Çelik has described two advantages to this approach: 1) It allows one to keep in touch with the many people who use popular social media platforms, 2) by aggregating “interactions from Twitter, independent websites, and Google+, all in one place” one “has an experience on his site that’s better than any silo” (Çelik 2014).

POSSE demonstrates a simultaneously dependent and antagonistic relationship between IndieWeb and corporate platforms, expressed in the proclamation, “We still use the silos as a distribution mechanism. We will use the tools of our competitors, of our enemies, to further our own causes” (Çelik 2014). Unsurprisingly, there have been cases where those “enemies” restricted types or quantities of data transmission in ways that compromised IndieWeb’s goals. This highlights the heterogeneous environment in which IndieWeb operates. I investigate this relationship through a study of Bridgy, a tool to syndicate content and responses between personal IndieWeb sites and platform APIs. This chapter describes Bridgy’s relationship with Facebook’s API from 2014 through 2018. This investigation is driven by two questions: (1) How might IndieWeb’s goals be challenged by its reliance on corporate APIs? (2) If problems arise, how are they addressed? These questions help address this dissertation’s overarching RQs

by identifying third-party platforms as centres of power that operate external to IndieWeb’s community, and nonetheless shape the way values are defined, selected, critiqued, and employed.

7.2 Using “tools of our enemies”

One of the biggest challenges for alternative social media is attracting users away from popular corporate platforms, which have enormous reach. For example, 84 per cent of online Canadians had a Facebook account in 2017 (Gruzd et al. 2018), and 79 per cent of online U.S. adults used Facebook in 2016 (Smith and Anderson 2018). Owing to this pervasiveness, there is little chance that emerging alternatives will displace corporate platforms any time soon. IndieWeb’s POSSE approach is an attempt to leverage the reach of large corporate platforms without becoming dependent on them.

A recurring theme during my data collection was a belief among IndieWeb’s contributors that corporate platforms should not be gatekeepers for communication.

I always use the telephone analogy [...] Even across countries and across companies you can have a phone number and we can still communicate with each other on a cellular phone network, but somehow the internet doesn’t work that way and it’s stupid that it doesn’t. Or even things like Twitter and Facebook, you should be able to talk to each other and have one account that does both and it just drives me bonkers that that’s not the case.

(Interview participant #13)

This demonstrates a widely held objection to the *platformization of infrastructure* and *infrastructuralization of platforms* described by Plantin (2017): “Google, Facebook, and a handful of other corporate giants have learned to exploit the power of platforms—which hold undeniable benefits for both users and smaller, independent application developers—to gain footholds as the modern-day equivalents of the railroad, telephone, and electric utility monopolies of the late 19th and the 20th centuries” (pp. 14-15). The consequence of this phenomenon is that platforms are increasingly as ubiquitous and essential as infrastructures, while “expression, communication, and knowledge are constrained within profit-driven corporate ecosystems” (2017, p. 3).

The POSSE model simultaneously acquiesces to and disrupts this phenomenon. When services like Bridgy automatically syndicate content from a personal website to a platform, they tighten bonds between the Web and corporate platforms. But from the perspective of people who use Bridgy on their own sites, POSSE offers a way to get the benefits of corporate social media while avoiding some of its downsides. Assessing this trade-off demands an understanding of how IndieWeb’s community understands the disadvantages of using corporate social media.

Most obviously, syndicating one’s content to platforms does little to mitigate surveillance capitalism (Zuboff 2019) in which corporate platforms generate profit by commodifying users’ personal data. I asked interview participants about their level of concern with platforms’ commodification of user data, and responses were mixed. In some cases, participants indicated that this wasn’t a personal concern for them but that it was relevant for others. This was usually qualified with a recognition of privilege:

I’m probably less concerned about that than most people. This is definitely due to privilege because I am not terribly concerned about being targeted. [...] To the larger community I think there are definitely cases where it can be hugely problematic. In that way it’s definitely

important that there is some level of control or at least awareness of what data is being used and gathered and by whom. (Interview participant #15)

This was not universal. One interview participant indicated that controlling how personal data is used is “absolutely crucial and very important for everybody on the IndieWeb as well” (Interview participant #5). In general, participants avoided collecting users’ data in software they created, but had a fatalistic attitude about corporate platforms’ commodification of user data:

I don’t think other members are that concerned about it. Granted, that’s probably because they have not seen a reliable alternative to these solutions. I think until we find something else, we’ll deal with this for now. (Interview with Jacky Alciné)

I also think that that is not something that is necessarily solvable because there are always going to be entities that want to collect and aggregate data for various reasons and will always find a way to do it. So, it’s a very very tough problem to solve. I’m not saying it’s not important, but [...] I think there are more problems that are not related to that that are worth solving first that will get us better bigger wins quicker. (Interview participant #15)

Instead, IndieWeb’s objection to corporate platforms is mostly based on how they restrict access to users’ content in “walled gardens” or “silos,” which makes individuals dependent on platforms to access their own content. This attitude is certainly evident in the overall framing of an *independent* Web and is articulated in design decisions about how to interact with platforms. For example, the lead developer of IndieWeb-friendly blogging platform Micro.blog expressed the following about the possibility of relying on third-party platforms to provide a login mechanism:

A lot of apps allow you to sign in with Twitter. In some cases, they require it. Like, to make it easy to create an account you sign in with Twitter and everything’s good. I will never do anything like that [...] I don’t want to depend on anything else for something so important as the user creating an account or creating a new blog. (Interview with Manton Reece)

In many cases, concerns about such platform sign-in features have focused on privacy and security (e.g. Egelman 2013; Saint-Louis 2018). That this is presented as a question of dependence is a reflection on a general attitude among IndieWeb’s community that lock-in or reliance upon corporate platforms should always be avoided. It is in this respect that POSSE subverts platformization by allowing individuals to access platform features without becoming dependent on platforms to access their content or data.

7.3 Syndicating to Platforms with Bridgy

The most popular tool for syndicating content between IndieWeb sites and platforms is a Web service called *Bridgy*. As described on its website, Bridgy “pulls comments, likes, and reshares on social networks back to your Web site. You can also use it to post to social networks - or comment, like, reshare, or even RSVP - from your own web site” (Bridgy 2020). Like many IndieWeb tools, Bridgy is provided for free and is funded out of pocket by its creator. Bridgy works by translating between popular platform APIs and IndieWeb sites that are built with standards like Webmention, Microformats 2 (MF2), and conventional Web standards including HTTP, URLs, and HTML.

When syndicating a post from one’s website to a social network using Bridgy, the post should be marked up using MF2 to help Bridgy identify the post’s content, type, and if it is a reply to someone

else. And when a response is *backfed* from a social network to a users' personal website, it is sent via Webmention. This means that a prerequisite for using Bridgy is a personal website that can receive Webmentions and uses at least basic Microformats.¹

When communicating with platforms, Bridgy uses their APIs. In contrast to the Webmentions and MF2, which Bridgy can expect to be relatively consistent across websites, APIs vary across platforms, each allowing access to different types of data and requiring distinct methods and identifiers to facilitate that access. Bridgy can connect to APIs for Twitter, Google+, Instagram, Flickr, and GitHub, though its features for each platform vary according to constraints of individual APIs. For example, Bridgy's documentation explains that Twitter's search API is "best effort only" so "every now and then [Bridgy] may not be able to find one of your replies" (Bridgy 2020). Significantly, Bridgy used to support Facebook, but this feature was shut down in August 2018 when, in response to the Cambridge Analytica scandal, Facebook dramatically restricted their API, removing capabilities that were necessary for Bridgy's function (this will be discussed further in this chapter).

Bridgy is one of the most widely used pieces of IndieWeb software. At the time I write this (June 16, 2020), Bridgy's website indicates that there are 4,566 user accounts (<https://brid.gy>). To put Bridgy's scope in context, a blog post published on January 11, 2018 to celebrate that one million Webmentions had been sent in the wild reported that Brid.gy had been used to send 960,778 of them. Moreover, Bridgy has been recognized as influential for opening up IndieWeb beyond its early adopters, since it was the first. It's been described in IndieWeb's wiki as "literally a bridge from Generation 1 to Generation 2² [...] because it allows Generation 2 to own their content on their own site, post to a silo, but also get the conversation from the silo posted back on their own site" (IndieWeb.org 2020b). Bridgy's popularity and influence make it a well-suited site through which to investigate IndieWeb's relationship with corporate platforms.

7.4 Data analysis

As with most IndieWeb projects, Bridgy is open source and hosted on GitHub, a platform for software code and other version-controlled repositories. GitHub repositories include a detailed history of revisions to source code and other documents, making it possible to identify how a project has changed over time. Individual changes are published as "commits" and discussion threads about new features and bugs are archived as "issues." Issues provide an apt entry point for investigating Bridgy's connections to other systems. First, they highlight breakdowns, since many issues literally describe errors, bugs, and other problems with Bridgy. Second, they present traces of the work that was conducted to resolve these breakdowns. In many cases, these discussions reference specific commits in which an issue was addressed or resolved. As a result, it can be possible to observe deliberation in developer discussions about issues, and then identify ways in which issues were addressed through code.

I downloaded copies of Bridgy's issues for analysis on March 5, 2018.³ At that time, there had been 2438 commits pushed to Bridgy's repository. Approximately 92 per cent ($N = 2238$) of commits were made by Bridgy's creator and lead developer, Ryan Barrett. One other user had contributed

¹The bare minimum is that posts are marked up with the *Microformats 1* `entry-content` property, which is widely used. Optionally, MF2 tags can be used to add more detailed information. (see <https://brid.gy/about#microformats>)

²As discussed in Chapter 6, generation 1 refers to highly skilled development leaders in IndieWeb's community, with generations 2-4 representing users with progressively more limited degrees of technical experience with Web development.

³The script used for downloading GitHub issues is available at <https://github.com/jackjamieson2/GitHub-issues-to-pdf>.

approximately 7 per cent ($N = 177$), and the remainder had contributed between 1 to 10 commits each. The earliest issue was posted on January 4, 2014. In total there were 799 issues posted between that date and the time of data collection. Across those issues, there were 4385 issue events (opening a new issue or commenting on an existing issue), contributed by 123 people had contributed. The median number of participants per issue was two, and the maximum was 8. This indicates that discussions around Bridgy’s issues were concentrated to a fairly small group overall (especially per individual issue), however the number of contributors is well above average when compared to the average of all IndieWeb-related GitHub repositories presented in Chapter 5.

Having collected the contents of Bridgy’s 799 issues, I performed a keyword search to get a general sense of how many issues pertained to each platform, with the following results: Twitter ($N=316$), Facebook ($N=278$), Google ($N=165$), Instagram ($N=99$), Flickr ($N=50$), Blogger ($N=32$).⁴ This is not an exact measure because the presence of a keyword does not guarantee that the issue actually pertains to that platform, and some keywords are ambiguous (e.g. “Google” can refer to a variety of products, and Bridgy only has an API relationship with Google+). Nonetheless, this preliminary query contributes a rough sketch of where Bridgy has had the most issues in relation to corporate platforms.

I selected issues involving Facebook for further analysis because (1) they represent a significant proportion of Bridgy’s total issues, and (2) Facebook’s large scope means it demonstrates many different types of interactions and potential problems. In an interview, Bridgy’s creator indicated that Facebook is a good representation of Bridgy’s relationship with corporate Web platforms (silos) more generally: “Any feature in any silo, Facebook has it too. And then they also have ten or a hundred thousand features that silo doesn’t. There aren’t many concerns that you don’t see in Facebook.”

A first pass reading of the 278 issues that included the term “Facebook” revealed that 147 of them described some sort of problem or feature request related to Facebook’s API. I then used open coding to organize the issues into categories that described types of problems that were most prominent. Where possible, I followed links to code commits that were referenced in the issue discussions, which assisted in identifying how Bridgy’s developers responded to different types of challenges.

While studying GitHub data, I was mindful for potential pitfalls presented by Kalliamvakou et al. (2016). While most of these pitfalls refer to large-scale quantitative studies, and so are not relevant to this study, I will address two pitfalls that needed to be considered. First, like many projects, Bridgy is not contained wholly within a single repository, instead relying on pieces from other repositories to perform some of its functions. Studying a single repository without considering its dependencies can lead to an incomplete understanding. A benefit of treating Bridgy’s issues as the entry for this study is that the issues posted to the main Bridgy repository often reference commits to its dependencies. Therefore, it was possible to start with Bridgy and expand outward to its dependencies, avoiding a potential blind spot. Second, GitHub projects almost always involve development and discussion occurring outside of the GitHub platform. In Bridgy’s case, this largely extends to IndieWeb’s chat channels and wiki, with which I have been engaged throughout this dissertation. In addition to the background and context provided by my other analyses throughout this dissertation, my investigation of Bridgy issues was supplemented with a semi-structured interview with Bridgy’s creator and lead developer, Ryan Barrett. This interview served to verify my assessment of the technical challenges and approaches as discovered through analysis of its GitHub repository, as well as to develop a richer understanding of Barrett’s motivations and

⁴At the time of this study, Bridgy had recently added compatibility for syndicating to and from GitHub. This feature was not included in this study because it was brand new and lacked a significant number of issues for analysis.

Table 7.1: Types of formats for Facebook IDs

Format	Example
“Simple number, usually a user or post”	12
“Two numbers with underscore, usually POST_COMMENT or USER_POST”	12_34
“Three numbers with underscores, USER_POST_COMMENT”	12_34_56
“Three numbers with colons, USER:POST:SHARD”	12:34:63
“Two numbers with colon, POST:SHARD”	12:34
“Four numbers with colons/underscore, USER:POST:SHARD_COMMENT”	12:34:56_56
“Five numbers with colons/underscore, USER:EVENT:UNKNOWN:UNKNOWN_UNKNOWN. Not currently supported!”	111599105530674:195181727490727: 10102446236688861: 10205257726909910_195198790822354

experiences.

7.5 Results

The issues reported on Bridgy’s GitHub reference a wide variety of topics. However, three types of problems emerged as the most substantial and recurring causes of breakdowns. Below, I summarize these problems as well as ways in which developers responded to them.

7.5.1 Mapping between URLs and API IDs

When syndicating between personal Web sites and Facebook, Bridgy spans a threshold between the open Web and platform APIs. The most prominent challenge in Bridgy’s development history has been translating between different ways of addressing an object across this boundary. Objects on the open Web are addressed using URLs, such as `http://facebook.com/{user-id}/{object-id}`, whereas the same object could be identified within Facebook’s API using an ID in a format such as `{user-id}_{object-id}`.

In some instances, it is straightforward to translate between these formats. In the example above, one can see how it could be possible to map between these identifiers using `{user-id}` and `{object-id}`. However, this mapping is often quite difficult for a variety of reasons. The first reason is that Facebook IDs come in several different formats, represented in Table 7.1 based on a comment in Bridgy’s source code.⁵ Bridgy’s developers struggled to predict a rationale for which format is required in different cases, and Bridgy’s code resorts to a sequence of guessing and trial and error to find the correct format.

Mapping between URLs and API IDs became more difficult in 2014, when Facebook released version 2.0 of its API. This update limited the amount of data third-party developers could access. In 2018, when Facebook CEO Mark Zuckerberg testified before the U.S. House of Representatives Committee on Energy and Commerce about Cambridge Analytica’s collection of Facebook users’ data, he asserted

⁵Table 7.1 is based on lines 1594-1614 at <https://github.com/snarfed/granary/blob/1e2568698de6f93ce383db2631cdfddfec9634e/granary/facebook.py>

that this update “makes it so a developer today can’t do what Kogan [the developer who shared data from his app with Cambridge Analytica] did years ago” (Zuckerberg 2018). One of the changes was the introduction of app-scoped user IDs, which mean that each third-party app is given a different ID for the same user. This improves Facebook users’ privacy and security because it frustrates efforts to combine data collected by multiple apps. However, this also complicates Bridgy’s efforts to map between a URL on Facebook.com and the corresponding object in Facebook’s API.

This led a notable reduction in Bridgy’s capabilities. Prior to this update, Bridgy could be used to like a Facebook post from one’s own website, without visiting Facebook directly. To do so, one would write a post to their website that linked to a Facebook post and included Microformats 2 markup to designate the post as a “like”. For example, a post might include the following HTML:

```
Alice liked <a href='{URL}' 'class='u-like-of'>a post on Facebook</a>.
```

Bridgy could interpret the post’s HTML and, if the URL points to a post on Facebook.com, could find the matching object in Facebook’s API and then post a “like” on the user’s behalf. The reason this is no longer possible is that URLs on Facebook.com refer to users by either their username or a global user ID, and neither of these identifiers can be mapped to the app-scoped ID Bridgy must use when navigating Facebook’s API. As a result, Bridgy’s developers removed support for this feature.

Although Bridgy’s ability to syndicate likes from a personal website to a Facebook account was removed, in most cases Bridgy has been successful at mapping between URLs and API IDs, albeit with considerable effort. From Facebook’s perspective, IDs are intended to be taken as opaque objects, and Bridgy’s attempts to decode them are not supported. Facebook support has advised Bridgy’s developers, “Please treat IDs as unique strings, they are not meant to be broken down and used” (Facebook for Developers 2019b). However, it is only when broken down and used that these IDs can be translated in URLs. To accomplish this, Bridgy relies on heuristics and trial and error, as explained by its lead developer:

There’s no consistent way, either through the API or through an algorithm you implement yourself, to say, ‘Here is a Facebook post, what is its permalink on Facebook.com?’ We have to guess at that with a surprising number of heuristics. Not ideal. So that mapping back and forth between the Web and data inside Facebook has been the single biggest question.

(Interview with Ryan Barrett)

Bridging the open Web with Facebook’s API necessitates reconciling different meanings and expectations on either side. The IDs used to reference content within Facebook act as boundary objects, serving as “a means of translation” from the IndieWeb to Facebook (G. C. Bowker and Star 2000), meeting local requirements while also maintaining a common identity. This allows different groups to work together in the absence of consensus by “tacking” back and forth between an object’s general form, which is meaningful across communities, and specific forms of the object tailored to meet local needs (Star 2010). Bridgy and Facebook share a general understanding of IDs as a means of referring to objects, but differ significantly in their local expectations of how IDs should be used. As a result, Bridgy uses heuristics, trial and error, and similar methods to fit the shared general understanding of IDs with its local requirements.

7.5.2 Mapping privacy from front-end to back-end

There are two main mechanisms for users to control who accesses their data on Facebook. First, they may use Facebook’s privacy settings to specify an audience that can view their posts and other information on Facebook.com or in the Facebook app. Options include “Public”, “Friends”, “Friends except...”, “Specific friends”, and “Only me.” Second, whenever a third-party app such as Bridgy wants to access Facebook data, users must grant it permission. For example, to access a user’s posts and photos, a third-party app will display a notice that the app will receive “your timeline posts and photos” — If the user agrees, the app will be granted the *user_posts* and *user_photos* permissions and will be able to access that data in Facebook’s API.

These two privacy mechanisms work independently. Notably, there is no way to grant a third-party developer access to public photos, but not to photos intended for friends only. As a result, it has been up to third-party developers to manage their treatment of posts with different audiences. This is significant for Bridgy because it processes only public data. When Bridgy retrieves data from Facebook to one’s personal website, it publishes a publicly visible copy of that interaction on Bridgy’s website. This is necessary to assign each interaction a URL, which is required to send a valid Webmention.

When accessing content in Facebook’s API, Bridgy checks its privacy status and ignores any content not marked explicitly as ‘public.’ Barrett commented on this process:

That is non-trivial to determine for a given object in the Facebook API, is it public? I mean usually, 90% of the time, it’s straightforward. Another 9% it takes some work, but you can figure it out, again depending on the type. It’s like 1% or maybe 0.1% where you actually can’t tell. You look at the inheritance chain, you look at a bunch of other stuff, and you just don’t know. If you go look at the UI [User Interface] in Facebook, you can usually tell. But programmatically you can’t. And so, when that happens, I have to err on the side of not doing anything with it.

(Interview with Ryan Barrett)

The most striking example of this difficulty occurs with some photos uploaded to Facebook. In Facebook’s API, photos themselves do not possess a privacy status. Instead, each photo is part of a parent post and/or album, which contains a privacy field indicating the intended audience.⁶ Therefore, determining the privacy status for a photo in Facebook’s requires one to check this parent.

When someone posts multiple photos to their Facebook timeline in a short period of time, Facebook creates a parent post representing all of them as a group, even if the individual photos have distinct privacy settings. In these cases, the parent post may have a privacy status of “CUSTOM”. In a GitHub issue, Barrett explained that Facebook’s documentation “[does not] say anything about what CUSTOM with no details means.”⁷ As a result, there are rare cases when it is not possible for Bridgy to determine the privacy status of an object in the API, even if it would be apparent through Facebook’s user interface.

When this was reported as a potential bug to Facebook’s developer support, the response indicated that Bridgy’s method of constructing the photo object’s ID resulted in accessing the photo through an unusual path where the privacy status could be unidentifiable. Moreover, the support representative indicated that one should *never* have to construct object IDs in this way, demonstrating that Bridgy’s usage was beyond Facebook’s expectations (Facebook for Developers 2019a). This highlights how unconventional efforts to repurpose platforms can surface hidden features of their systems.

⁶The privacy field for an object in Facebook’s specifies a privacy setting among these options: EVERYONE, ALL_FRIENDS, FRIENDS_OF_FRIENDS, SELF, CUSTOM.

⁷See <https://github.com/snarfed/bridgy/issues/611#issuecomment-174315265>

To avoid accidentally publishing non-public data, Bridgy’s approach is to ignore any data that is not explicitly marked as public. This has led to cases where Bridgy fails to process content that is intended to be public if the privacy setting in Facebook’s API is unclear. This has been interpreted by multiple users as a bug on Bridgy’s part, since the data is marked as public in Facebook’s user interface and yet was ignored by Bridgy.

This example helps position the importance of privacy in Bridgy’s design. IndieWeb is generally less concerned with privacy than some other alternative social media, as evidenced by practices such as syndicating content to corporate platforms and the fact that IndieWeb sites are almost always publicly accessible and indexable by search-engines. However, even though Bridgy does little to enhance individuals’ privacy, it is designed to avoid infringing upon existing privacy expectations. Shilton et al.’s (Shilton, Koepfler, and Fleischmann 2013) sociotechnical dimensions of values provide axes for describing the place of privacy in Bridgy’s design: Saliency (peripheral to central), intention (accidental to purposive), and enactment (potential to performed). In many use-cases, privacy is a potential rather than performed value in Bridgy, since the software usually deals with public data. When there is a potential infringement, however, it becomes clear that privacy has a high saliency and intention, since it supersedes Bridgy’s proper functioning. The result is a value dam (J. K. Miller et al. 2007) where privacy is so central to Bridgy’s lead developer that he opposes conflicting designs, even when it limits the software’s perceived efficacy.

By identifying this breakdown empirically, I have surfaced privacy as a salient value for Bridgy’s developer, even if Bridgy’s overall design is not privacy focused. A photo’s privacy status was only ambiguous in a specific configuration of relations in which Bridgy was accessing that photo using a method that has been inferred experimentally rather than based on documented features of Facebook’s API. Only when Bridgy’s approach met specific conditions—*the photo being posted in a particular fashion that affected how Facebook’s API recorded its privacy status*—did a value of *privacy* become a suitable hypothesis for resolving the situation at hand.

7.5.3 Precarity and API updates

APIs can change quickly and unpredictably. While this study was underway, Facebook issued substantial API updates to improve its security and privacy in light of the Cambridge Analytica scandal. These updates removed the ability for third-party apps to publish content to one’s Facebook account (Archibong, 2018), which meant that Bridgy could no longer syndicate from one’s website to Facebook. A subsequent update introduced restrictions that limited Bridgy’s ability to send comments and likes from Facebook back to one’s website. Prior to this, once a user had authorized Bridgy to access their posts using Facebook’s API, Bridgy was able to fetch all responses to a users’ posts. After the API update, Facebook would only allow Bridgy to access responses posted by other Facebook users who had also authorized Bridgy to access their account. This meant Bridgy could no longer be used to backfeed responses from people who were not also Bridgy users, which constituted “probably 98-99%” of interactions Bridgy had been backfeeding from Facebook (Barrett 2019). Since this closed off the vast majority of Bridgy’s Facebook’s functionality, Bridgy dropped support for Facebook altogether (Barrett 2018).

Until 2018, the precarity of Bridgy’s relationship with Facebook was somewhat subtle. For the most part, Bridgy’s developers had been able to maintain functionality with Facebook’s API, albeit with considerable expertise and effort. By studying Bridgy’s development history, this study was able to illuminate the labour of responding to API updates, and particularly the decision-making to preserve

values such as privacy. As reliance on third-party APIs has become a prominent feature of software development, the impact of API updates has been investigated in several studies (Espinha, Zaidman, and Gross 2014; Hora et al. 2015; Xavier et al. 2017). Nonetheless, excepting those who experience bugs or participate in development processes, the precarity of Bridgy and other software that rely on third-party APIs is often hidden.

During our interview, Barrett was quick to assert that problems he encountered with Facebook’s API were not a result of malice. Rather, “in many ways what Bridgy is doing is not at all what Facebook expects the average Facebook app to do.” Further, he asserted that an app like Bridgy is simply too small to be a concern for Facebook, citing his past work experiences as a senior engineer at Google: “I have seen some it at the inside of Google. For things that aren’t security breaches—you know for apps that are just doing funny things that may or may not be against your TOS [terms of service]—if they’re small enough you don’t care.” The development history of Bridgy indicates that Facebook and similar platforms are not actively attempting to prevent Bridgy’s style of syndication. In fact, in the past Facebook has solicited work on a similar tool from software developer and open Web advocate Dave Winer (Ingram 2014a). In this case, it is unlikely that anyone at Facebook was specifically motivated to restrict services like Bridgy. Instead, Facebook’s attitude toward Bridgy could be characterized as indifference.

This indifference creates opportunities for experimentation and innovation, but as demonstrated by Facebook’s dramatic API updates in 2018, also cultivates substantial risk. Another IndieWeb developer who created a commercial service with similar syndication features as Bridgy has written that reliance on platform APIs had been an obstacle, especially from a business perspective, “Back when I was working on Known, investors would ask about the supplier risk of being so heavily dependent on third party APIs to provide a lot of the core value. They were right” (Werdmüller 2018).

One of the goals of this study was to consider how skilled developers might have agency to contest platform features with which they disagree, rather than simply opting out or proposing wholly separate alternatives. Although Bridgy was, for a time, successful at navigating its relationship with Facebook, the pressure of the Cambridge Analytica scandal prompted a set of restrictions in Facebook’s API that Bridgy could not accommodate. While the circumstances surrounding Facebook’s API restrictions were dramatic, they represent a precarity that extends to all similar dependence on platforms. Nonetheless, the purpose of POSSE in the first place is to reduce dependence upon platforms by encouraging people to post to their personal website first, rather than allowing the only copy of one’s content to be on a platform. In this respect, the outcome when Bridgy has lost access to platforms demonstrates resilience of this overall approach. For example, Bridgy also lost support for Google+ in March 2019, when Google closed its API in preparation for shutting down all personal Google+ accounts in April 2019. When Google announced that it was shutting down Google+ for personal accounts, it included the following warning:

On April 2nd, your Google+ account and any Google+ pages you created will be shut down and we will begin deleting content from consumer Google+ accounts. Photos and videos from Google+ in your Album Archive and your Google+ pages will also be deleted. (Google 2019)

Those who had been using Bridgy to POSSE from their website to Google+ did not risk losing their content, since they had never depended on Google to host it in the first place. Further, amidst this

discussion of Bridgy losing functionality as a result of API closures and changes, I would like to acknowledge that as of the time of this writing (in summer of 2020), other platforms through which Bridgy supports POSSE remain functional (e.g. Twitter, Instagram, Flickr, GitHub) and new platforms have been added, including Meetup, Reddit, and Mastodon. In essence, platforms targeted for POSSE are re-configured as part of IndieWeb’s network of *building blocks* to be “adopted individually in combination” (IndieWeb.org 2019c).

7.6 Discussion and conclusion

By investigating challenges in Bridgy’s development from 2014 through 2018, I have illuminated consequences of IndieWeb’s position within broader Internet systems that are increasingly structured by corporate platforms. I will conclude this chapter by returning to the research questions posed at its beginning, and then discussing implications of the results for the IndieWeb and design research more generally.

This chapter asked the following questions: (1) *How might IndieWeb’s goals be challenged by its reliance on corporate APIs?* (2) *If problems arise, how are they addressed?* The results of this analysis show that Bridgy’s dependence on Facebook’s API presented two general types of obstacles. First, Bridgy’s success in repurposing Facebook’s API as part of a distribution architecture required consistent maintenance. This included managing differing logics for identifying and addressing resources while working across the open Web and platforms. In Bridgy’s case, these logics could usually be managed through heuristics and other forms of articulation work. Additionally, difficulty determining the privacy status of some objects led to an ethical dilemma. Although this second sort of problem was relatively rare, it surfaced a limitation in Facebook’s design that would not occur during conventional use. And further, it surfaced a commitment in Bridgy’s design to preserving privacy, where privacy was not otherwise a significant priority in Bridgy’s overall design. Instances of this first type of obstacle could never be resolved outright, but instead required ongoing maintenance and repair. Second, and more straightforwardly, all of Bridgy’s interactions with Facebook were contingent upon its API preserving a base set of functionalities. When Facebook dramatically reduced that functionality in 2018 it was no longer deemed possible for Bridgy’s developers to maintain support for Facebook.

Bridgy’s *raison d’être* is to mediate between systems, and accordingly it explicitly demonstrates *artful integration* focused on “the collective achievement of new, more productive interactions among devices, and more powerful integrations across devices and between devices and the settings of their use” (Suchman 2002, p. 99). Bridgy’s utility exists only in relation to actors such as its users and the systems with which it operates, and is demonstrated through the practices it enables among them. This does not indicate that Bridgy exemplifies a special and unusual type of design that relies on networks of relations, but rather that it explicates the role of such relations, which are mutually generative for all objects. This perspective of design as existing within networks of mediation is a counter to the stance of “design from nowhere,” which “is closely tied to the goal of construing technical systems as commodities that can be stabilized and cut loose from the sites of their production long enough to be exported en masse to the sites of their use” (2002, p. 101). When Bridgy’s use of Facebook’s API achieved stabilization, it was a mutual accomplishment based on interactions between the sites. Yet, since Facebook’s developers were essentially indifferent to Bridgy’s purpose, the burden of maintaining that accomplishment fell to Bridgy’s developers, as Facebook’s API changes and other inconsistencies

regularly contributed to destabilization. This demonstrates that Facebook’s API acted as a technical tie (Damarin 2013) that influenced the labour of building Bridgy. Insofar as this influence was not reciprocal, and it is unlikely that IndieWeb is a significant influence on the maintenance labour of any large corporate platform, POSSE positions corporate platforms as centralized points of influence, albeit mostly limited to interactions in which they materially participate.

With regard to values, I will briefly discuss the challenges for preserving users’ privacy that emerged from the specifics of Bridgy’s unconventional way of mapping resources in Facebook’s API. In this instance, privacy became observable as a value to be defended when it was at risk. I propose there is a meaningful distinction between values to be defended through design compared to values to be enhanced beyond their status. Many instances of values-oriented design take a critical approach based on challenging the status quo or addressing a perceived problem (e.g. Brunton and Nissenbaum 2015; Flanagan, Howe, and Nissenbaum 2005). In fact, this the basis of IndieWeb’s (and Bridgy’s) overall approach—to build an alternative way of using the Web that enhances individual autonomy and empowerment. By contrast, Bridgy’s approach to privacy is not an effort to disrupt a status quo. Specifically, Bridgy does not enhance individuals’ privacy compared to more typical uses of Facebook, but instead took action to *avoid deteriorating* Facebook’s existing privacy features.

Design strategies for defending values cannot be fully articulated in advance of use, since threats to values may only emerge as a result of unforeseen relationships. In essence, defending values may not be a case of attempting to stabilize a material artifact in a particular direction, but rather to identify and respond to emergent destabilizations. Addressing unanticipated threats to values can be served by Houston et al.’s (2016) proposed agenda to move away from a notion of *values* as universal and toward an understanding of *valuation* as “contingent, ongoing processes through which things are rendered valuable in a wider social and material context” (p. 1412). One of the consequences of this shift is an awareness that “what values are materialized and how they are made visible are deeply intertwined issues” (Houston et al. 2016, p. 1412).

This creates a point to return to this dissertation’s research questions. The questions asked in this chapter, about how developers’ goals may be challenged during interoperation with third-party systems, emphasize the link between defining and employing values (RQ1 and RQ2) that has been coming into focus across this dissertation. It is through their employment as meaningful hypotheses to serve specific situations that values become visible features of design. And thus, while some values are defined and prioritized in advance of material engagements, others may be taken for granted or otherwise invisible until a situation emerges that demands attention to a particular value.

The obvious power differential between Facebook and developers of software that accesses Facebook’s API affects how such situations are defined. Specifically, if certain values only become visible when a situation demands, then whoever can shape the situation influences how values are made visible, defined, and employed among all actors. Insofar as corporate platform shape much of the ground (e.g. the API) upon which POSSE unfolds, their design can invoke situations that bring certain values to light, while obfuscating others. Ultimately, this is not a determinative relationship, evidenced by Bridgy’s hard-won success in using Facebook and other platforms in ways they have not been intended. However, the fact that Bridgy had to use undocumented workarounds to navigate between URLs and objects in Facebook’s API increased the risk of violating a commitment to privacy that had been largely taken for granted—that Bridgy would avoid diminishing existing privacy protections by only processing public data. When Facebook’s API, in specific circumstances, did not provide some objects’ privacy statuses

to Bridgy in a consistent fashion, a value that seemed to have been adequately addressed came into question. The burden of resolving the resulting privacy risk fell onto Bridgy's developers, and it is only through attention to maintenance that the such emergent threats to values can be addressed.

This highlights the extent to which third-party apps are contingent upon the platforms they use. Since platforms generally follow the Web 2.0 model of constant iterative development (O'Reilly 2007), third-party developers are required to do the same. This means constantly update their software to maintain compatibility where possible, and dealing with the fallout of incompatibility where necessary. Platform instability can shape the viability of specific values as hypotheses during maintenance, since the situations to which values are brought to bear are in flux. In addition, ever-changing platform APIs are a technical tie that shapes the types of labour processes that are suitable for third-party applications. Specifically, this limits the potential for building long lasting software that relies on these platforms. Projects such as Bridgy are only possible if the developers can remain active over several years of maintenance. Especially in the case of unpaid software development, this is only possible for developers who are sufficiently committed and privileged to balance this labour against professional, family, and other obligations.

In closing, given the extent of platformization, many systems and tools must co-exist with corporate platforms to some extent. This is the basis of POSSE, which attempts to repurpose corporate platforms as distribution infrastructures. In this chapter, I have identified the extent to which articulation work is necessary to maintain the stabilization of such arrangements. Further, I have shown that situations that emerge through this work can necessitate the employment of values that had not previously been made visible, and I have argued that this is particularly important for *defending* values against destabilization. These are important considerations for IndieWeb's efforts to provide an alternative that operates alongside and in collaboration with corporate platforms while remaining independent of them.

Chapter 8

Discussion

The purpose of this dissertation has been to investigate the activities involved in building and maintaining IndieWeb, with particular attention to designers' values.

As I summarized in Chapter 2, much of the research about values and design emphasizes designers' decision-making. Yet it is clear that access to decision-making is not universal. I have used the concept of divisions of labour to acknowledge that decisions in one location can result in labour processes that constrain possibilities for decision-making elsewhere. In the absence of a managerial hierarchy, I have adopted Damarin's (2013) *network-organized labour process* theory to build a conceptual framework describing how divisions of labour are communicated through technical, human, and postsocial ties. The impact of this approach is that addressing designers' values requires accounting for centralizations and decentralizations of power and acknowledging that these lead to exclusion and prescribe some decisions unto designers as matters of fact.

To that end I have asked the following three research questions:

- **RQ1:** What values are central to the IndieWeb, and how are they defined, selected, and critiqued?
- **RQ2:** How are values employed in design activities?
- **RQ3:** Is power over planning and executing design activities centralized or decentralized among stakeholders, and is anyone included or excluded as a result?

I have addressed these questions throughout Chapters 4 to 7. I will briefly summarize the results, in the process explaining how they relate to one another.

I will then discuss some challenges and insights that arose from my use of multiple methods as well as from my approach to building IndieWeb software. In this description I will identify where I encountered epistemic clashes and how I have attempted to reconcile them. I close this chapter with a brief essay identifying key implications of this dissertation for studying design. Specifically, I explain how this work contributes insights that can serve an agenda where HCI and related design approaches are focused on nurturing and sustaining human well-being.

8.1 RQ1 and RQ2

In this section, I will return to RQ1 and RQ2:

- **RQ1:** What values are central to the IndieWeb, and how are they defined, selected, and critiqued?
- **RQ2:** How are values employed in design activities?

In addressing these questions, what JafariNaimi, Nathan, and Hargraves (2015) term the *identify/apply* logic common to values and design research was well-suited to describing situations in which a specific type of practice was codified, such as in IndieWeb’s principles. However, their view of *values as hypotheses* was advantageous for identifying cases where prominent espoused values were insufficient for responding to particular situations. This perspective was particularly useful for making sense of circumstances where *processes* of design demonstrated shifting values, even when working toward a consistent outcome.

Further, I have highlighted throughout the results of this dissertation that the employment of values in design activities is one of the processes through which values are defined, selected, and critiqued. Accordingly, I will summarize RQ1 and RQ2 in tandem. This brief review will be structured as a list of values that have been identified as important for IndieWeb’s design, each accompanied by a brief explanation.

Individual autonomy and Self-empowerment: I identified these as explicit, stated values in Chapter 4. These were encoded in IndieWeb’s principles for building, both as outcomes to be achieved (e.g. *own your data*) and as guidelines for building processes (e.g. *Build what you need; pluralism; modularity*). These values were defined in advance of the IndieWeb, inherited largely from early members’ past Web experiences. As outcome-oriented values, these resonate with commitments of other decentralized Web projects (see for e.g. Kahle 2015; Berners-Lee 2016). The extent to which these values were employed in processes of building was presented as a novel feature of the IndieWeb, exemplified by “encouraging everyone to build what works for them, and then figure out how to interoperate between different coding/implementation approaches” (IndieWeb.org 2018b). In this respect, IndieWeb demonstrated a commitment to material-empowerment that is common in making communities (Roedl, S. Bardzell, and J. Bardzell 2015)—i.e. that building for oneself was framed as a pathway to empowerment and autonomy. However, through my own experiences building I observed a balancing act between autonomy and commensurability, which is best addressed through explanation of *benevolence*, *inclusion*, and *conformity* values below.

Benevolence values were discussed in Chapter 4 as expressed in norms of sharing documentation and code as well as building in a modular style so that other contributors could re-use sections of one’s code. These were explicated as values for defining the activities around building. In the case of designing code that could be re-used by others, benevolence was also articulated as an outcome of IndieWeb software and other artifacts. Additionally, taken for granted norms about collaboration and discussion (discussed in Chapter 4) and influence articulated through knowledge sharing (discussed in Chapter 6) demonstrate the importance placed on supportive social relations among IndieWeb’s community members.

Inclusion was identified as particularly challenging in Chapter 6. While this was discussed primarily in terms of RQ3 (i.e. identifying sources of exclusion), inclusion was evident as a value in several features of IndieWeb’s community, such as its code of conduct and diversity initiatives. I also noted in Chapter 4 that my effort to design for inclusion through *ease-of-use* was challenged by IndieWeb’s modular structure. Further, much of the apparent exclusion I described in Chapter 6 was related to early *development leaders’* decisions, which contributed to a material architecture that served their interests at the expense of accessibility for people with less technical aptitude and/or interest.

Privacy was conspicuously absent as a central value for IndieWeb’s design, which I addressed briefly

in Chapter 4. While privacy was recognized as an important value by many contributors, it was also regarded as a low priority because it was difficult to achieve. I argued that, even though autonomy and privacy are closely aligned as outcomes (e.g. in controlling access to information about oneself), as a value for defining building *processes*, autonomy engendered a heterogeneous approach that precluded the coordination required to build a system that focuses on enhancing privacy. Privacy came up again in Chapter 7 when IndieWeb’s public-by-default architecture intersected with Facebook’s privacy controls. I observed a situation in which, although privacy was not apparent as part of a software artifact’s design, a situation arose that demanded attention to privacy, not as a target for enhancement, but to avoid causing harm. This result supports the need to attend to *valuation* as a process (Houston et al. 2016), since unenacted values may become visible as conditions in and around a system change.

Conformity values were identified in Chapter 4 as expressed through technical standards and collaborative processes, and in Chapter 6 as expressed through community-scale policies and initiatives. Conformity was only observed as a value for guiding building processes, typically aiming to achieve a requirement for consensus in a given situation. For example, conformity articulated through standards served interoperability among a plurality of personal websites, facilitating individual autonomy and empowerment. Where standards captured existing best-practices, the act of codifying those practices encourages (and sometimes demands) conformity moving forward. This is precisely the point of a standard, and standardization itself is such an entrenched feature of software engineering that it is difficult to imagine complex systems that are not governed by technical specifications. Cases where standards were implemented from the top down (e.g. the introduction of the Microsub specification before there were any implementations) seemed to indicate situations where values focused on individual autonomy had been insufficient for guiding building processes. Similarly, when initiatives to improve inclusion were implemented at the community level, they identified similar limits to what could be achieved through individualist values. The visibility of conformity as a value in both IndieWeb’s technical architecture and community policies was somewhat surprising given IndieWeb’s overall emphasis on individual freedom. However, conformity was always a means unto that end rather than an end itself.

Having reviewed these major values, I will review IndieWeb’s overall political nature. By focusing on small, iterative, and personal interventions, IndieWeb attempts to carve out spaces for alternative practices within the existing Internet, rather than attempting to replace it. In this respect, IndieWeb’s commitment to internal plurality—i.e. that its contributors should be encouraged to build multiple solutions that can co-exist—is mirrored in its relationship to external systems. Most clearly, the POSSE¹ model explicitly supports the ability of IndieWeb sites to co-exist alongside the corporate Web, as discussed in Chapter 7. In Chapter 1, I drew on Atton (2002) to make a distinction between oppositional and alternative practices. To return to this concept, this dissertation has demonstrated IndieWeb to be more alternative than oppositional, evidenced by its purposeful co-existence with the corporate web and its contentment with local and personal accomplishment rather than sweeping reform.

This scope is related to IndieWeb’s commitment to building for the present as a means to address personal, immediate needs and its generally self-managed structure. To some extent, this emphasis on individuals building for their own needs echoes the libertarian ethic of the Californian ideology (Barbrook and Cameron 1996). This ideology has been a defining feature of Silicon Valley’s tech industry, including elements with which IndieWeb is well-aligned (i.e. open Web oriented companies and organizations) and with which it is in tension (i.e. corporate Web platforms). It is no surprise then, that IndieWeb has

¹*Publish on Own Site, Syndicate Elsewhere*

routinely been described as libertarian or overly individualist by critics. For example, a critique in early 2020 described IndieWeb as “right-libertarian” and “individualist rather than collectivist,” (Belshaw 2020). In response, Çelik stated that “‘libertarian’ (any kind) is a misimpression at best. #IndieWeb succeeds & grows because it’s a #community co-operating for individual agency” (Çelik 2020b). He then drew upon another account of IndieWeb as demonstrating an “inclusive and really anarchistic punk vision of a web for all, where everyone can DIY and co-create the future” (rabble 2019 as cited in Çelik 2020a)² My observations about IndieWeb’s employment of conformity and benevolence values to shape its processes affirm this rebuttal. Even though individual autonomy is a central motivation for the IndieWeb, I argued in Chapter 6 that inclusion is a strong priority and is most successfully achieved through collective interventions. Further, as I have argued in Chapter 4, although technical work tends to begin at the scale of individuals serving their own needs, this is routinely overtaken by communal approaches to achieve interoperability and consensus.

8.2 RQ3

In this section, I will return to RQ3:

- **RQ3:** Is power over planning and executing design activities centralized or decentralized among stakeholders, and is anyone included or excluded as a result?

By distinguishing between *planning* and *executing* design in this question, I used the concept of division of labour to consider how concentrations of influence may operate with regard to both the definition of values and other goals and the actual work of building IndieWeb software, websites, and other artifacts. Of course, the reality is that planning and execution are not performed in isolation, but instead that building activities are situated actions (Suchman 2007) that can acquiesce to, contradict, or bypass plans. As noted in Chapter 2, IndieWeb lacks a formal hierarchy or other structure through which *plans* are prescribed unto developers. To that end, this discussion of centralization and decentralization of power will be organized according to three types of influence as drawn from Damarin (2013): *Technical ties*, *human ties*, and *postsocial ties*.

8.2.1 Technical ties

I identified a number of technical ties in Chapter 4. First, I identified a balancing act between technical features that support decentralized building activities (e.g. modular and pluralistic designs) and those necessary to achieve interoperability (e.g. technical standards). The modularity and plurality of IndieWeb’s building block approach was presented as a method of enhancing individuals’ control over what and how they built. In many respects this was successful, but I also noted that modularity itself was a technical tie that shaped my development processes. Technical standards were a significant technical tie, insofar as they literally prescribe a number of design decisions. It was usually the case, that technical standards were written to capture and codify existing practices rather than prescribing new practices. When this was achieved, it enhanced the influence of early implementors whose decisions could become imbricated in subsequent standards, as well as established as norms even if not explicitly codified. Additionally, technical ties were influential as sources of inspiration. I experienced this personally

²Since this quote was posted by Çelik without linking to a primary source, I verified its accuracy with @rabble on Twitter.

when working on Yarns and several interview participants described being influenced by others' work. These observations helped establish technical implementation as the most visible way for individuals to influence other developers.

This contributed to a concentration of influence among those with technical ability and motivation to implement IndieWeb software and share code, which was a major point of discussion in Chapter 6. Among members of this group, however, influence was bidirectional because developers were often aware of each other's work and sought to maintain compatibility. Among users of IndieWeb software who were not developers, technical ties enacted a firmer influence, more or less determining what options were available for engaging with the IndieWeb, though users always had a choice among multiple options. Further, although those who considered themselves to be less technologically adept may have felt less influential, there was a degree to which they could still be influential by identifying challenges that others had not experienced and otherwise broadening the diversity of needs that were acknowledged by IndieWeb's community.

By contrast, technical ties arising from outside IndieWeb exhibited unidirectional influence, as was discussed in Chapter 7. When IndieWeb sites were bridged with corporate platforms, the platform APIs structured most of the situation in which interaction occurred. This was not a total domination, since IndieWeb developers were able to repurpose corporate APIs beyond their intended purpose, but possibilities were sharply constrained by decisions made in the API's design. Unidirectional influence can result in fatalistic attitudes, such as some remarks about POSSE in the wake of Bridgy's discontinuation of Facebook support, and could limit perceived utility of certain values (e.g. privacy) in situations where their achievement seemed excessively difficult.

8.2.2 Human ties

Human ties were a significant influence on my development activities in Chapter 4, including through direct recommendations or suggestions about how I should proceed. Most dramatically, I adopted Microsub in part based on encouragement from other IndieWeb members. This was gentle encouragement, and had I been opposed to the suggestion would not have swayed me. Less dramatic, though more substantial, were ongoing processes of troubleshooting through discussions with other developers, especially those who created software with which Yarns interacted. Beyond my own experience, interview participants identified conversations with other IndieWeb members as impactful for learning how to set up their websites or solve coding problems (in Chapter 6). In this respect, human ties exhibited significant influence over how technical artifacts were designed.

In general, influence from human ties was decentralized, occurring through person-to-person interactions. However, given the importance of online chat discussions to this end, the concentration of chat and other online activities identified in Chapter 5 is significant for describing the role of human ties. Unsurprisingly, a relatively small core of extremely active users was influential over online discussions and development activities. With regard to how this centre was defined, I described how technical ties oriented around modularity and plurality were easy to navigate for people who most closely resembled IndieWeb's earliest adopters, who tended to be technically skilled developers. I also noted that technical barriers for participation intersected with barriers related to gender, race, and other dimensions of social privilege. I argued in Chapter 6 that addressing these problems was best served at a community level, which required coordination among community organizers.

The distinction between human and technical ties is somewhat blurry, at least in that they often occur

together. In the case of many technical artifacts internal to the IndieWeb, encounters with the artifact itself are often accompanied with at least the possibility of encounters with its creator. When technical ties are encountered without access to a human representative, as in most of Bridgy’s encounters with Facebook’s API, they project influence but do not receive it. In essence, the possibility of negotiation with human ties softened the influence of technical ties. When one has the opportunity to engage with an artifact’s maintainer by asking questions, reporting bugs, or requesting features or other accommodation, there is a possibility to interject in technical ties that might otherwise appear fixed. In this respect, IndieWeb’s emphasis on community, particularly through events and open online discussions, is a force of decentralization when it comes to technical influence, even accounting for its apparent centre of active members.

8.2.3 Postsocial ties

Postsocial ties are a deliberate absence when it comes to individuals’ building activities. IndieWeb’s rationale for building for one’s own real-world needs discourages designing for typified others: “If you design for some hypothetical user, they may not actually exist; if you make for yourself, you actually do exist” (IndieWeb.org 2020i). When postsocial ties emerged as influences in Damarin’s (2013) research, they were associated with economic motivations, such as designing for “the market” or generalized representations of “clients” and “end-users” (p. 194). In the predominantly non-commercial context of the IndieWeb, it has been viable to attempt to avoid such influence in individual design activities.³ However, even though I generally followed IndieWeb’s principle of building for my own needs while I worked on Yarns, postsocial ties were an influence in my effort to make Yarns easy to use for a typified class of non-technical user. It is likely that *users* as postsocial ties were similarly present in other developers’ projects, though this was not voiced during interviews or otherwise surfaced as a prominent theme.

However, postsocial ties were prominent when it came to conceptualizing IndieWeb more broadly, especially with regard to IndieWeb’s future. Most explicitly, the *generations* model used to describe IndieWeb’s growth relies on generalized representations of future users. In this model, “development-leaders” and “people using social networks” are framed under the same umbrella of “potential IndieWeb adopters” rather than being classified as developers and users (IndieWeb.org 2020b). Since IndieWeb’s principles and practices encourage avoiding generalized conceptions of hypothetical end-users, this provides a useful vantage to identify other types of postsocial ties. Specifically, it is clear that IndieWeb’s contributors have generalized concepts of *other community members* that influence their decisions and perspectives. One aspect of this was identified through criticism of the *generations* model in Chapter 6, including that some people have attempted to classify themselves as members of a particular generation, and that this can articulate a rhetorical divide that limits their sense of mobility. Additionally, as I reflect on my work building Yarns, some of the influence that I attributed to human ties and technical ties was also filtered through my sense of what *IndieWeb developers* do. Most specifically, I adopted Microsub largely because it was an emerging norm, something I believed *other community members* would take up, and thus a good tool through which to understand IndieWeb’s community. To the extent that IndieWeb’s community members are influenced by generalizations about their peers, this somewhat

³In this dissertation, I have focused on non-commercial instances of IndieWeb development, though I acknowledge that there are exceptions to this approach. I have briefly touched upon commercial IndieWeb endeavours (*Micro.blog* and *Known*) but I have not described their development in depth.

intensifies the influence of IndieWeb’s most vocal participants.

Finally, the strongest typified other among IndieWeb contributors is articulated through reference to the *silos*, the corporate platforms against which IndieWeb is largely a response. As an influence upon the IndieWeb, generalizations about the silos function more to rally people together than to urge specific design decisions. Specifically, when IndieWeb software engages with the silos (most obviously through POSSE), they are encountered as technical rather than postsocial ties. In fact, the material rather than theoretical engagement demonstrated through POSSE is what engenders the ability to re-purpose platforms, treating them as contestable “matters of concern” rather than immutable “matters of fact” (see both Latour 2004; Ratto 2011).

8.3 Working across scales and methods

When presenting the research design in Chapter 3, I acknowledged that the multiple methods used in this dissertation demonstrate epistemic clashes. These can be conceptualized as *epistemic double-binds* in which epistemic commitments of multiple disciplines come into conflict (Ratto forthcoming; Ratto 2016; Ratto 2009). To describe this, Ratto (forthcoming) and Ratto (2016) draws on Agre’s (1997b) account of experiencing vertigo as he began to incorporate critical approaches alongside technical work. Specifically, Ratto notes that epistemic difficulties from crossing interdisciplinary boundaries are often products of our own patterns of thinking that have been instilled through disciplinary training. I experienced two distinct boundary crossings during my research, which I will discuss here.

First, the analysis of IndieWeb’s structure that was presented in Chapter 5 diverges significantly from the types of analysis elsewhere in my dissertation. In large part, this dissertation has attempted to follow Suchman’s (2002) recommendation to take a situated perspective of design. To that end, I have focused on the activity involved in building and maintaining IndieWeb’s system and attempted to resist the impulse toward a ‘view from nowhere’ (Haraway 1988). That is, I have consciously avoided viewing IndieWeb as a universal and stable technology (or even agglomeration of technologies) that can be packaged for tidy delivery to new contexts. To serve that commitment, I have emphasized ongoing processes of maintenance, drawn from my own subjective experiences of IndieWeb development, and considered IndieWeb’s community as integral for understanding its material features.

And yet, I’m not sure I can think of a more literal representation of a view from nowhere than the network analyses and statistical topic models I used to describe IndieWeb’s community in Chapter 5. A myriad of rich and nuanced discussions is flattened into a short list of topics. Relationships formed through the exchange of ideas, explanations of technical processes, debates, jokes, and other conversations are measured by quantity, rather than quality, of discussion. And similarly, the hard work of making design decisions, writing code, and troubleshooting bugs on GitHub is reduced to a description of frequency and types of interactions. For my research questions, which are largely about questions of values that are inherently nuanced, these representations are on their own insufficient.

Nonetheless, they have also been valuable for extending my analysis of IndieWeb’s community to a broad scale. Rather than relying on quantitative measures, it would have been possible, albeit time-consuming, to have simply read IndieWeb’s chat logs from start to finish. And indeed, I have spent a great deal of time reading those chat logs. However, realistically, I could not have achieved a large-scale analysis of GitHub activity without computational methods. Every step of that process, from identifying IndieWeb-related repositories on GitHub, to retrieving the logs for each repo, and finally to analyzing

that data was automated through Python scripts. In order for the chat and GitHub analyses to be commensurable, their data had to be similarly structured. Ergo, quantitative methods facilitated an analysis of IndieWeb's online activities at a scale that would have otherwise been impractical.

To attend to this clash, I have approached the structural results of Chapter 5 as only part of the story. I have limited the scope of my interpretation of those results, identified their inability to fully address RQ3, and accompanied them with a more intimate perspective in Chapter 6. Most of all, I have attempted to describe the processes involved in this dissertation's quantitative analyses in sufficient detail to show that they are in reality profoundly situated, just at a different vantage point than is achieved through interviews or ethnographic perspectives. This results in triangulation by viewing the same phenomena through multiple methodological lenses, each with their own theoretical apparatus (see Denzin 2015). While these pose an epistemic clash, in so doing they reveal a productive tension through which I have enriched the scope and substance of this dissertation. In sum, these analyses bring important insights. In fact, the most important contribution of Chapter 5's analysis has been to help me situate my personal experiences and other observations within an understanding of IndieWeb's scope, something that would have been difficult to achieve without this crossing.

A second tension was evident in the process of building software for IndieWeb. My motivation was to understand building processes, which is a far cry from IndieWeb's emphasis on building something useful to oneself, or the importance interview participants attributed to building and sharing code. To qualify that difference, I'll first acknowledge that (1) Yarns was designed to address my own needs, and has resulted in an artifact that is useful for me, and (2) that by building Yarns I have engaged in a practice of writing and sharing code, though at limited scale that is not particularly influential. However, the most valuable part of Yarns for this dissertation was the experience of constructing it, which cannot be communicated by sharing code.

In fact, for all the time invested in working on Yarns, the actual software will not be evaluated as an indicator of my scholarly qualifications. My committee members and other readers of this dissertation are likely to only encounter Yarns as a *textual doppelganger* (Ratto 2014) in the form of its description in Chapter 4. It is unlikely that my committee members would use Yarns or review its source code as part of this dissertation's assessment and doing so would not capture Yarns's value anyway. As a part of this doctoral work, Yarns was valuable in that it provided a means to enter the process of building IndieWeb software, to experience influence extended through relationships with other IndieWeb building blocks, and to note the extent to which building software to serve my own needs became intensely social.

Overall, there was some tension between my approach to Yarns as a vehicle for reflection and its utility as a piece of software. I discussed this in Chapter 4 by describing how material structures that facilitate modular code contributed to obstacles for reflection. In this instance, experiencing this tension was beneficial because it drew my attention to ways in which norms about development practice (both in a general sense and related to IndieWeb's conventions) influenced my decision-making. This helped structure my thinking, especially about *technical ties*.

8.4 Independent Together

The final interview for this dissertation was held in a bustling cafe a short walk from the pre-summit social at IndieWeb Summit 2019. The participant (Interview participant #3) and I wrapped up the interview just in time to walk over to the social, and we continued our conversation while we walked.

A topic came up that I regrettably have not had opportunity to cover in this dissertation: *How would IndieWeb handle a situation where its building blocks are adopted to communicate harassment or hate speech?* This has been a major concern among corporate social networks, with companies like Facebook and Twitter devoting increasing attention toward content moderation (Gillespie 2018). The response typically employed by corporate platforms, to hire moderators to review and remove offensive content, is not feasible in a system like IndieWeb that lacks centralized gatekeepers by design. Another interview participant had remarked on this while reflecting on an incident where he had received racist messages on Twitter:

I couldn't use Twitter properly for like three days. I'm wondering now, had I been using the IndieWeb then I would have had no way to moderate my site. I would have had to do it manually. Like, every mention that came in. It was like hundreds of people.

(Interview with Jacky Alciné)

IndieWeb does have some technical solutions in progress for addressing spam and harassment, most visibly the *Vouch* extension for Webmention,⁴ but they are generally unproven because spam and harassment have generally not occurred through Webmentions or other IndieWeb tools.

Beyond technical solutions like Vouch, my final interview participant suggested other ways that IndieWeb could mitigate hate speech and harassment. First, creating an IndieWeb site generally requires an effort (to set up and operate the site) and money (to pay for hosting), which would discourage the use of disposable online personas often used for trolling. In brief, the investment required to adopt IndieWeb's approach encourages people to treat their IndieWeb identity as something substantial, not worth tarnishing by being a troll. Second, if someone came to IndieWeb's community asking for help setting up a site or tool that would facilitate hate speech or harassment, they would be violating IndieWeb's code of conduct, asked to leave, and certainly not assisted. This is a reasonable argument when it comes to IndieWeb's present, though I am not convinced it will apply with regard to IndieWeb's future. It is likely that it will become easier, more frictionless, and cheaper or free to access IndieWeb-compatible Web hosting of some description, and it is entirely possible that could lead to more spam, harassment, and other antisocial communications.

Nonetheless, this highlights (1) that a high barrier to entry, at least relative to corporate platforms, can be a virtue, and (2) the extent to which IndieWeb's community is a core component of its infrastructure. As I have discussed in Chapter 6, IndieWeb's apparent barrier to entry has often been regarded as a problem, or at best a necessary compromise. This was the first argument I encountered from one of my interview participants that IndieWeb's barrier to entry could be a virtue unto itself. This has led me to reflect on a complementary argument presented by Turkle (2011) in her book *Alone Together*. Turkle's thesis is that by embracing the ease, convenience, and control offered by easy communication methods like text messages and social media, we have sacrificed the substance of conversation. A major part of the lure is that we simply don't have to invest in those kinds of connections in the same way we do when conversing face-to-face or even on the phone. The result, she argues, is that we are connected but isolated, hence *alone together*.

IndieWeb does not solve this problem. There is little reason to believe that a Webmention inherently engenders conversation in a way that a Twitter mention or Facebook comment does not. However, the collective effort of building IndieWeb's infrastructure, and the fact that it is a *collective* accomplishment,

⁴See <https://indieweb.org/Vouch>

contribute to greater levels of substance. Borgmann's (2010) argument that engagement with reality is commodified through technology provides a conceptual foothold here. In the same way Borgmann argues that nuanced labour, expertise, and experience are flattened through commodification, part of the *alone together* phenomenon is that rich dimensions of human relationships are commodified through technological means of connection. This sort of commodification exerts a soft determination upon the sorts of engagement made possible through communication technology and is central to IndieWeb's objection to corporate platforms. Especially in my discussion of Bridgy's relationship with Facebook's API in Chapter 7, it is clear that responses to this objection are not all-or-nothing. Here, again, I will refer to Borgmann to frame the importance of a middle ground:

If you order your food from SeamlessWeb (it delivers a ready-to-eat gourmet meal to your doorstep) and eat it while surfing the internet, you are in the thrall of technology. If on a late afternoon you and your children go harvesting in your vegetable garden and if in the evening you prepare the meal with your spouse and sit down to dinner with your beloved, you are blessed. If on the way home you pick up prepared food at the store, warm it, wait for your spouse, and sit down to eat it with her, you are on the side of the angels. (Borgmann 2010, p. 34)

Applied to IndieWeb, I would mirror this statement to suggest the following: If you spend your day posting to Facebook, Twitter, TikTok, Snapchat and other corporate platforms, you are in the thrall of technology. If you spend a weekend at an IndieWebCamp collaborating with like-minded peers on building your own system to replace corporate social media, you are blessed. If you read a friend's post in a Microsub reader and post a thoughtful response to your carefully constructed personal website, then use Bridgy to syndicate it to your other friends on social media, you are on the side of the angels. Though admittedly, these blessed and the angels here are more aligned with the thrall of technology than in Borgmann's example, and I suspect he might view this comparison with skepticism.

The recovery of engagement possible here resonates with Roedl, S. Bardzell, and J. Bardzell's (2015) claim that

research about “maker culture” is frequently framed as a challenge to HCI's traditional conception of the “user.” This discourse is also often accompanied by claims that the maker, as a specific configuration of a technological subject, is particularly well positioned to bring about increased democracy and empowerment. That is, the argument goes that if HCI acknowledges the DIY movement and begins to design for “makers” instead of “users,” then the field will help to further “empower” and “democratize” society (p. 2).

This conditions around this claim were discussed in more detail in Chapter 4, but I will also note here that Roedl, S. Bardzell, and J. Bardzell (2015) conclude that the potential social good of focusing on *makers* is contingent upon addressing challenges far beyond the technical, such as legal, physical, economic, and other constraints, as well as social issues such as racism, sexism, and other forms of injustice outside the traditional purview of HCI. Dourish (2019) suggests that one of the obstacles for addressing such challenges in HCI is that the field has staked its claim to legitimacy in the concept of user experience (UX). The boundaries of what can be addressed through UX are visible in Hassenzahl and Tractinsky's (2006) proposed research agenda for UX: “From our perspective, one of HCI's main objectives in the future is to contribute to our quality of life by designing for pleasure rather than for absence of pain. UX is all about this idea” (p. 95). By contrast, Dourish (2019) asserts that “the central charge to HCI is to

nurture and sustain human dignity and flourishing”—a distinctly more ambitious goal than designing for pleasure—and asserts, “By legitimizing HCI and its role in technology production in terms of user experience, user delight, and user acceptance—which were only ever means toward other ends—we have ceded the space from which we could argue for the considerations that were actually at the center of the discipline’s ambitions.” Ultimately, this legitimacy trap scopes HCI researchers and designers’ attention to the design of products, systems, and other material goods. Dourish introduces the concept of UX as a legitimacy trap to complement related work in HCI, calling out critical HCI (J. Bardzell, S. Bardzell, and Blythe 2018) and value-sensitive design (Friedman and Hendry 2019) as similarly aligned to his endeavour.

In the remainder of this chapter, I will briefly argue that my dissertation contributes to this toolset, and in doing so explores a design context that resists the impulse to commodification (Borgmann 2010) and isolation (Turkle 2011), instead emphasizing ongoing and communal achievement. I will frame this argument with the two insights identified during my final interview, *investment as a virtue* and the role of IndieWeb’s *community as part of design*.

8.4.1 Investment as a virtue

The first insight noted above is that the need to invest time and effort into one’s IndieWeb site (or other projects) can be regarded as a virtue. One aspect of a requirement for investment is simply that it filters out people who are not willing to make the right kind of effort, or who lack certain kinds of resources. This type of filtering was clearly at work in IndieWeb’s early period, where people who were not actively building for the IndieWeb were excluded by event policies and technical requirements for RSVPing. I’d also like to consider if requiring investment can nurture substantial forms of engagement, pushing against the forces of commodification that arise when human experience is mediated through technological *products* or when design is viewed in the service of pleasant or delightful user experiences. To present a meta-level reflection, the example that comes to my mind is that many parts of writing this dissertation have been neither pleasant nor delightful, but instead challenging and even frustrating. However, this dissertation is nonetheless an experience I would not trade for something easier. What does it look like to design for that kind of experience?

To discuss this, I will return to IndieWeb’s modularity. And to address that modularity in the context of UX as a legitimacy trap, I’ll briefly refer back to IndieWeb’s principle #7: “UX and design is more important than protocols, formats, data models, schema etc. [...] AKA UX before plumbing.” (IndieWeb.org 2020i). One pattern that has become clear is that, although plumbing might come second to UX, IndieWeb features a lot of exposed plumbing as a result of its modularity.

Is this good UX? If one’s view is that good UX entails a positive emotional experience while using a product, then this is clearly not good UX for everyone. Chapter 6 included several accounts where IndieWeb’s modularity and the visibility of its plumbing was a source of confusion and discouragement to certain people. Yet, if designing for good *developer experience*, modularity and exposed plumbing are advantageous. This is what happens when one designs for makers (or in this case, when makers design for themselves), a different set of priorities emerge than when designing for “users.”

Perhaps a more significant ramification of modularity, especially accompanied with its principle of plurality, is that IndieWeb is never presented as a stable technical artifact. I specify *presented* as stable here because stabilization in corporate platforms, too, is always temporary (Humphreys 2005). Probably the clearest example that IndieWeb’s stabilization is impermanent was in Chapter 7’s description of

Bridgy’s breakdown and repair due to corporate API changes. Studying these processes highlighted that breakdowns were usually accompanied by opportunities to find a clever solution (excepting, of course, major reductions in API function, which rendered a more permanent sort of destabilization).

That unstable state where there is room to negotiate or figure out a way forward extends across IndieWeb more generally. Ultimately, insofar as IndieWeb’s building blocks and software are perpetually extensible, there is never a point in which discussions, debates, and relationships among community members are congealed in a stable material outcome. This is not to say that IndieWeb developers have access to interpretive flexibility over every part of its system. I have noted that early technical decisions tend to become imbricated over time, and certain features require compromise between individual autonomy and conformity. Nor should one conclude that instability is a simple good, since it does contribute to exclusions on the basis of technical aptitude and interest, and these exclusions should be taken seriously. However, where there is a capacity for interpretive flexibility, there are both opportunities and requirements for decision-making. But nonetheless, I argue that IndieWeb’s design articulates its meritorious outcomes—such as autonomy, inclusion, and empowerment—not as things to be given to a user, but as an achievement that requires ongoing work to maintain. When a system demonstrates instability, it invites people in to participate in its interpretation and direction.

In affirming the importance of ongoing investment in technologies, my work reaffirms recent scholarship that has emphasized treating values as processes (Houston et al. 2016), hypotheses (JafariNaimi, Nathan, and Hargraves 2015), or otherwise attending to values as something other than static concepts to be applied to design. Further, my analysis supports attention to values in other stages of technological development, such as pursued by Whittle (2019):

Values-based methods are well known in human–computer interaction (HCI) and information systems but are nonexistent in software engineering. HCI and information systems do not deal with the business of actually building software, so although they could apply in the early stage of software engineering, they offer little guidance as to how to handle values in the more technical stages of development (p. 114).

Although many activities involved in constructing the IndieWeb can be understood within the purview of UX design and related aspects of HCI, much of the hard work occurred through later stages of maintaining code, managing interoperability, and other later stages of development. Had I not developed software myself nor studied breakdowns by reviewing discussions and source code in Bridgy, I would have presented a very different perspective of what building for the IndieWeb entails.

8.4.2 Community as part of design

My conceptual framework in this dissertation has focused on empirically observed articulations of values. From this perspective, it would be possible to focus only on values that are explicitly tied to systems design, leaving aside values in IndieWeb’s community and social structures as external to design. Doing so would have been consistent with the legitimacy trap identified by Dourish (2019). However, because I have sought to identify ways in which IndieWeb’s sociotechnical infrastructure could centralize or decentralize decision-making structures, I have attended closely to its community. Further, even when I have been focused on apparently technical features of the IndieWeb, such as its modularity, I have found these to be tightly intertwined with community structures.

It is, of course, possible to adopt IndieWeb standards or tools without otherwise engaging with the IndieWeb community, but the abundance of IndieWeb events and meetups, as well as its active online discussion spaces, demonstrate that discussions with IndieWeb contributors are near ubiquitous. Even newcomers typically encounter IndieWeb through events and online discussions or at least turn to those sites for advice and guidance. In this respect, it is clear that IndieWeb's community not only shapes its material outcomes but is in fact part of them.

One of the influences upon my thinking about community and organizational culture in design was danah boyd's (2019) acceptance speech for the 2019 Barlow/Pioneer Award. In this speech, boyd reflects upon the unjust structures through which she has navigated, and ultimately thrived, to reach that point in her career. She described her time at MIT Media Lab as "full of contradictions" with memories of "laughter and late night conversations" alongside witnessing "so much harassment, so much bullying that it all started to feel normal." She remarks on similar experiences throughout her career, and notes that John Perry Barlow, the namesake of the award she was receiving, once asked her to forgive a friend of his whose sexual predation she had witnessed. She refused. And she addressed a scandal that was unfolding at MIT Media Lab at the time:

I'm receiving this award, named after Barlow less than a week after Joi [Ito] resigned from an institution that nearly destroyed me after he socialized with and took money from a known pedophile. Let me be clear — this is deeply destabilizing for me. I am here today in-no-small-part because I benefited from the generosity of men who tolerated and, in effect, enabled unethical, immoral, and criminal men.

boyd argues, and I strongly agree, that values and ethics must be taken seriously throughout an organization's processes, and this must occur even when they challenge an organization's financial or other functional outcomes.

If change is going to happen, values and ethics need to have a seat in the boardroom. Corporate governance goes beyond protecting the interests of capitalism. Change also means that the ideas and concerns of all people need to be a part of the design phase and the auditing of systems, even if this slows down the process.

I highlight boyd's speech because it elucidates the ethical stakes of organizational culture. Tech, both corporate and non-corporate, has routinely presented itself as creating innovations that make the world a better place. This claim suffers when the cultures of so many technical organizations reproduce and even amplify injustice. The most important takeaway from boyd's speech is that the field of technology, like many others, cannot move forward without reckoning with injustices in which it has been complicit. I believe that part of this reckoning will involve designers and researchers taking seriously that organizational and community cultures do not merely contribute to but are part of design outcomes.

How then, can communities oriented around technology, especially those with commitments to human dignity and flourishing, engage in this reckoning? Taking seriously the role of values can be helpful here. However, empirical research about values and design, including my own investigation in this dissertation, cannot see beyond specific actions, objects, or situations. In other words, looking at moments of design in which someone demonstrates a commitment to a value such as inclusiveness or individual autonomy may suggest that someone holds that as a value in other aspects of their life, but this can't be known without observation. In fact, boyd's critique is based on her observation that people and organizations who have upheld positive values in some aspects of their work have disregarded those values elsewhere.

She highlights the costs of letting ends justify their means—of looking away from wrongdoing for the sake of building a great product, getting research funding, or growing one’s company.

One way to address this is to broaden the scope of what one considers to be design work. I have argued that IndieWeb’s community is as much a part of its design as its material systems. Communities like IndieWeb should be aware of this, and I have highlighted in this dissertation that IndieWeb’s organizers demonstrate and act upon such awareness. Further, I have argued that IndieWeb is not directed toward a stable or permanent outcome, but instead it is an ongoing, continuous, collective accomplishment. In this respect, there is not a coherent way to distinguish ends from means. When one has put that distinction aside, it becomes clear that one should strive for consistency in the values they bring to bear across all aspects of an endeavour. This is not straightforward in two respects. First, people have unconscious biases and taken for granted predilections that can make it difficult to notice when their values fall out of step with their actions. Second, when attempting to build something new, we rely on existing tools, infrastructures, and concepts, which may direct us back toward the problems we are working against. For example, it is clear that IndieWeb’s use of technical standards is in tension with its commitments to individual autonomy and plurality, insofar as technical standards are inherently about conformity.⁵ And, as I argued in Chapter 6, commitments to inclusion are in tension with requirements for specific types of technical aptitude, which intersect with various forms of privilege. In practice, such tensions cannot be avoided altogether, and new tensions will emerge continually as infrastructures shift over time. This should never be taken as an excuse for compromising one’s values, but instead contradictions should be brought to light. Exposing and engaging with value tensions provides an opportunity for members of a community to share responsibility for maintaining the integrity of their endeavour.

8.4.3 Conclusion

Throughout this dissertation, I have taken seriously IndieWeb’s claim to be pursuing self-empowerment and autonomy, and I believe it is fair to extend Dourish’s (2019) charge of *nurturing and sustaining human dignity and flourishing* as a fair description of IndieWeb’s purpose. My intention here is not to claim that IndieWeb holds the key to achieving that purpose (I have noted both successes and failures throughout this dissertation). Instead, I have argued that my analysis of IndieWeb’s building activities contributes to efforts to extend the scope of HCI and design research to attend to this goal. In part, this is a support for an assertion observed in Roedl, S. Bardzell, and J. Bardzell’s (2015) review, that designing for makers rather than users can contribute to empowerment and democratization. My research has supported this thesis by identifying how IndieWeb developers often prefer systems with exposed plumbing and opportunities for intervention, even when this detracts from ease-of-use. In such cases, a lack of stability becomes an opportunity to enact change, thus encountering technology not as a fixed commodity but as a set of relationships among components that can be modified, moved, or replaced.

To address human dignity and flourishing, however, it is not sufficient to merely build flexible or adaptable technologies. It is also necessary to address social relations around the technology, including attention to past injustices and forms of exclusion. Chapter 6 discussed that IndieWeb has, like most communities oriented around technology, struggled to attract a culturally diverse group of contributors. Especially given IndieWeb’s emphasis on building for one’s own needs, the composition of its community

⁵This tension is explicitly addressed by IndieWeb’s commitment to small, modular building blocks, rather than a single IndieWeb standard, as I discussed in Chapter 4.

directly shapes what kinds of software and tools are pursued. Nonetheless, it is apparent that IndieWeb's organizers are attentive to this problem. This is not to say that every intervention was successful, but there is strong evidence that IndieWeb's community culture is acknowledged as equal to its material artifacts.

Chapter 9

Conclusion

In this dissertation I have investigated the activities involved in building and maintaining the IndieWeb, a network of personal websites connected by standards and software that allow them to engage in online social networking without being dependent on corporate platforms. My goal has been to investigate the role of designers' values in IndieWeb's development, including how concentrations of power and influence affected individuals' capacity to participate in decisions that include their values. This work has attended to concerns in HCI and STS, especially concerned with the role of values in technological systems. Methodologically, I have drawn on material engagement through critical making as well as reviewing others' code and development decisions, computational analyses of IndieWeb's online community interactions, and in-depth interviews with IndieWeb contributors. I have identified important values for IndieWeb's community of developers and observed how they are employed in design decisions of various types, including in response to challenges that emerge from interoperability within and external to IndieWeb's architecture.

I will conclude by summarizing contributions of this dissertation then identifying directions for future research.

9.1 Contributions

9.1.1 Values in HCI and design

This dissertation contributes efforts to conceptualize values as features of design. By investigating values in ongoing processes of extending and maintaining the IndieWeb I have identified ways they are employed in changing circumstances. By taking a longitudinal view I have been able to observe processes of destabilization. This has contributed to my understanding of the articulation of values in design as an achievement to be continually renewed and maintained, as well as illuminating how previous decisions and conditions shape who has access to make design decisions, and what kind of decisions are viable. Further, situating individual design activities as part of IndieWeb's community, as well as in relation to corporate Web platforms, has highlighted how influence over design extends through technical, human, and postsocial ties.

By observing design and maintenance in changing circumstances, I was able to present a distinction between values to be *defended* and values to be *enhanced*. While design strategies for enhancing a value tend to be planned in advance, the conditions in which values are defended may not become visible until

a threat emerges. To address this challenge, designers need new strategies to forecast and respond to threats. This is likely to be especially relevant in studies of interoperating systems, since functionality in one system is shaped by a third-party system that may be beyond one's influence.

Additionally, I have investigated IndieWeb's community not simply as a context that affects its design, but as integral part of its outcomes. IndieWeb's combination of a visible and engaged community alongside a modular and flexible material infrastructure creates both opportunity and responsibility for IndieWeb contributors to engage with its continual reproduction, including the ability to shape its direction. I argue this is a step toward extending HCI's conventional focus on UX, which limits its capacity to address substantive questions of justice and wellbeing in human relationships with technology (Dourish 2019).

9.1.2 Decentralized Web and alternative social media

Decentralized infrastructures have become prominent among emerging internet technologies. Among many types of decentralized architectures there are open questions about governance, and my analysis has presented IndieWeb's distinct model for balancing individual autonomy with commensurability. Further, I have engaged with Schneider's (Schneider 2019) assertion that architectural decentralization is typically accompanied by centralization elsewhere. Specifically, my analysis has highlighted that features oriented toward decentralization and plurality of development processes have contributed to a concentration of influence among early development leaders and created barriers for some newcomers. Further, I have identified incidents where conformity values were employed to create conditions that supported autonomy, sometimes as part of a counterbalance to the concentration of power that resulted from architectural decentralization.

Further, this dissertation generally addresses Gehl's (2015b) call for researchers to study alternative social media, since this can contribute to new ways of thinking about the effect of media infrastructures. While IndieWeb has been addressed in previous scholarship (e.g. Guy 2017) my study is the first I am aware of to present an in-depth analysis of its development. Further, IndieWeb's syndication with corporate platforms to which it is an alternative makes it a particularly valuable example of how alternative social media can evolve alongside the corporate web.

9.1.3 Logged social media data analysis

To my knowledge, Chapter 5 includes the first analysis of its kind that investigates relationships between developer chat participation and development and discussion on GitHub. This presents an important consideration for scholars studying GitHub as a social network (e.g. Kosner 2018; Strzalkowski et al. 2019). Additionally, insights about linking data between online sites can contribute to future research about how media multiplexity (Haythornthwaite 2001) operates in contemporary online communications.

Finally, this computational analysis was integrated alongside qualitative ethnographic approaches used elsewhere in this research. The ways I have identified and addressed resulting epistemic clashes, as well as the insights born from this combination, contribute to ever-changing approaches to digital ethnography.

9.1.4 IndieWeb

Finally, I hope this dissertation is a benefit to people in IndieWeb's community. I have identified relationships among IndieWeb's social structures, design decisions, material architecture, and connections to corporate platforms, which may be helpful for future decision-making. As a material contribution, this dissertation has included the first attempt I am aware of to develop a thorough list of IndieWeb-related repositories on GitHub. I will release this to IndieWeb's community, to whom I believe it can be a useful dataset. And I hope that Yarns can be a small-but-useful part of IndieWeb's system.

9.2 Future directions

Based on the results and implications of this dissertation, below are some areas to which I plan to attend in my future scholarship:

- One of the blind spots of this study was about the potential for so-called *bad actors* to use decentralized infrastructures for harassment or other malicious behaviour. This was briefly mentioned in Chapter 8 but there were very few reports by interview participants of such behaviour, and none was observed personally. In my future work, I will investigate contexts where open-source and decentralized social networks are used in ways that conflict with the values of their creators. This will extend a concept I have identified in this dissertation where defending values in design demands a different set of strategies than enhancing values.
- One of the reasons I elected to study IndieWeb is that it is particularly well-suited for an analysis of values. Even though IndieWeb does not draw from value-sensitive design or similarly formalized methodologies, it was clear from the outset that values were a priority among many of its developers. With regard to my observations that values extended to decisions about design and community policies in much the same way, this may be related to the general predisposition of IndieWeb's contributors. Specifically, it is worth considering how organizational cultures and design decisions relate to one another in commercial contexts, which are likely to firmly differentiate between developers and users as a result of their business priorities. Within the context of the IndieWeb, there is opportunity to study commercial projects such as Known and Micro.blog. More significant is the need make organizational culture a routine consideration during design inquiries, as this will help avoid circumstances where organizational values and norms conflict with design goals.
- As well as investigating the design of emerging decentralized and alternative social media, it will be important to consider how they may serve a variety of stakeholders and purposes. Gehl (2015b), for example, argues that alternative media organizations and activists should adopt alternative social media instead of relying on corporate platforms, since alternative infrastructures may be more axiologically aligned to their goals. Given IndieWeb's potential to support user/developer experiences based on achievement rather than conventional UX emphasis on pleasure or positive affect, there is an opportunity to consider how infrastructures like this can serve pedagogical goals. I have begun considering this question through work with Greg McVerry (McVerry and Jamieson 2018) and there is a need to continue this inquiry.
- The computational methods used to investigate chat and GitHub activities may be extended to be useful in other contexts. I would like to clean up and release my scripts for parts of this analysis

that may be useful for other researchers.

- Similarly, I plan to release data collected about IndieWeb’s chat and GitHub activities in a format that can be explored by IndieWeb’s community. If possible, I would like to build this into a Web application whose data can be regularly updated. Although all of this data is sourced from public archives, I will consider the potential risk of releasing it in this new context. Part of this consideration will be a discussion with IndieWeb’s community to understand their perspectives and concerns.

Appendices

Appendix A

Interview Invitation

In cases where I had interacted with interviewees prior to sending this invitation, this text was modified to acknowledge those past interactions. Similarly, many letters were customized to identify specific projects the participant had worked on about which I wished to ask. These modifications did not change the overall substance of the letter, such as the description about the nature of the study and the structure of the interview.

Dear **[Name]**,

My name is Jack Jamieson and I am a PhD candidate at the University of Toronto. I am conducting a study that examines the work of developers who have contributed to the IndieWeb.

I am interviewing people who have contributed code or design to projects related to the IndieWeb. Because of your work related to **[list of related projects]**, I am writing to ask if you are willing to be interviewed for this study.

My study is particularly concerned with experiences making technologies that reflect the designers' values (i.e. principles), and with issues around building and using alternatives to centralized and/or corporate platforms.

To address this subject, I would like to ask you about goals and motivations for contributing to software and/or design, and successes and challenges related to achieving these goals.

The interview can be scheduled at your convenience, and is expected to last about one hour. The interview can be held by video chat, telephone, or another method.

Attached to this email is a PDF detailing your rights if you choose to participate in an interview. / Here is a PDF detailing your rights if you choose to participate in an interview: http://jackjamieson.net/informed_consent-IndieWeb_interview.pdf ¹

If you have any questions, please contact me by email. You are also welcome to contact the University of Toronto Human Research Ethics program at 416-946-3273.

If you are willing to be interviewed for this project, please reply to this **[email/message]** or contact me at jack.jamieson@mail.utoronto.ca

¹Sent attachment if contacting by email; link if contacting by other messaging medium.

Your participation will contribute to the creation of knowledge about this important subject. Thank you very much for your time and consideration.

Regards,

Jack Jamieson

PhD Candidate, Faculty of Information, University of Toronto

jack.jamieson@mail.utoronto.ca

Appendix B

Interview Consent Form



UNIVERSITY OF TORONTO
FACULTY OF INFORMATION

Informed consent form:
The IndieWeb from values to execution

I have been informed about the nature of this study by Jack Jamieson from the University of Toronto, and I understand the purpose of the study as well as its potential benefits and/or risks to me. Based on this, I voluntarily consent to participate in the study, and agree to the following conditions:

- Freedom of withdrawal: I understand that my participation is voluntary and that I may withdraw from the study at any time. I can withdraw by informing the interviewer at any point during the interview. If I withdraw, I can choose not to allow any information already collected to be used in the study.
- I understand that I do not have to answer any questions during this interview if I do not want to.
- I understand that this interview will be recorded.
- Freedom to refuse participation: I understand that if I chose not to accept this agreement, no data will be collected and I will not be allowed to participate in this study.
- Rights to anonymity: I understand that I may choose to remain anonymous, in which case my identity will be protected in all reports and publications based on this research. I will be asked to indicate my preference at the beginning of the interview.
- The computer files storing the recordings and transcripts of the interview will be encrypted, password protected, and kept in a secure location. If applicable, any hard copy documents containing my personal information will be kept in locked storage. Only members of the research team, working directly with the dataset and bound by a confidentiality agreement, will have access to the data.
- The use of data: With these safeguards in mind, I agree that the interview can be used in reports, publications or presentations. In the case that I want to have access to the summary of the research results, I may contact Jack Jamieson at jack.jamieson@mail.utoronto.ca.
- The research study you are participating in may be reviewed for quality assurance to make sure that the required laws and guidelines are followed. If chosen, (a) representative(s) of the Human Research Ethics Program (HREP) may access study related data and/or consent materials as part of the review. All information accessed by the HREP will be upheld to the same level of confidentiality that has been stated by the research team.

If you have any questions, please contact Jack Jamieson (Faculty of Information, University of Toronto):
jack.jamieson@mail.utoronto.ca

Or his supervisor, Professor Rhonda McEwen (Faculty of Information, University of Toronto):
rhonda.mcewen@utoronto.ca

Or, if you have any questions about your rights as a participant in this research, you may directly contact the Office of Research Ethics at the University of Toronto at ethics.review@utoronto.ca or 416- 946- 3273.

If you choose to participate, you will be asked to confirm that you understand and agree with the points stated above at the beginning of the interview.

Thank you,
Jack Jamieson
PhD Candidate, Faculty of Information, University of Toronto | jack.jamieson@mail.utoronto.ca

Appendix C

Interview Protocol

Informed consent

- 1) Have you read the informed consent agreement, and do you have any questions about its contents?
Remind the participant that s/he can refuse to answer any question, and can exercise the option to withdraw from this interview by notifying me at any time.
- 2) You are given the option to remain anonymous, in which case documents related to this research will refer to you with a pseudonym **(or without any title)**. Would you prefer to be referred to using a pseudonym or your real name?

Pseudonym Real name

Remind the participant that they may change their mind at any time during this research, in which case they may notify me by email or other means.

Note: If you like, I can send you any quotations I use from this interview before I publish, so you can verify you are quoted accurately.

Wants to receive quotations | Doesn't want to receive quotations

- 3) Please confirm whether you agree to the conditions described in the consent form.

Introductory questions

- 4) What IndieWeb-related activities do you perform? (e.g. contributing to software or standards; posting on IndieWeb's IRC)
- 5) About how many hours per week do you spend on IndieWeb-related activities?
Includes design/code, writing about IndieWeb, and chatting on IRC etc.
- 6) Where and when do you usually work on IndieWeb projects (e.g. evening at home, office during work hours, etc.)
 - a) *Are there any factors that prevent you from working on IndieWeb projects as much as you would like?*
- 7) **What is your day job? Is it related to the work you do with the IndieWeb?**

- a) Can you talk a bit about how your day job(s) have related to IndieWeb over the past few years?

Probes:

- *Does your work experience contribute to your ability to work on IndieWeb projects?*
- *Does IndieWeb fulfill a different need than your professional work?*
- *Are you able to work on IndieWeb projects during your work day?*

Community structure questions

- 8) How did you first get involved with IndieWeb?
- 9) Are there specific people who have had large influence on your thinking about IndieWeb?

- 10) Do you think you have influenced IndieWeb's direction?
 a) *If so, how? (technical, principles, culture, other?)*
- 11) Have members of IndieWeb's community motivated you to change the direction of your projects?
 (e.g. to change design patterns, take on a new project, cancel a project)
- 12) Have you had disagreements with members of IndieWeb's community? (e.g. IRC, wiki, in-person events)
 a) What was the argument about?
 b) What happened?
 c) How was it resolved?
- 13) Is there anything you wish were different about IndieWeb?
 (1) *Disagree with direction (cultural, technical, etc.)*
 (2) *Do you ever feel intimidated or excluded?*
 b) If so, do you feel like you can influence this?
- 14) Is there anything you think could make this community more open and accessible for you or for other people?
 a) E.g. gender, race, culture, expertise

Questions about the definition of values by individuals and among IndieWeb's community:

- 15) What motivates you to participate in these projects?
 a) Are these motivations fulfilled?
- Probes:**
- Make money?
 - o *Do you think it's okay for IndieWeb projects to make money?*
 - Make something useful for yourself?
 - Have fun?
 - Advance career?
 - Make the world a better place?
- 16) Do you believe the design of the Web is an ethical issue? If so, what are the ethical stakes?
 a) If so, how should the Web work to be more ethical?
- 17) Next, I'd like to list a number of different goals that might be pursued by building standards or other tools (i.e. through building a tech stack). I'd like to tell me two things about each:
 a) Whether this goal is important to you.
 b) Whether you believe it is important to IndieWeb's community.
- 1- Not important
 - 2- Somewhat important
 - 3- Very important
 - X – no opinion or don't know

	Important to me	Important to IndieWeb community
Owning my data (and ask what this means)	1 2 3 X	1 2 3 X
Controlling where and how your posts and other published content is displayed to other platform/web users	1 2 3 X	1 2 3 X
Controlling how my personal data is used by web platforms (e.g. demographic data, 'likes', contact list, GPS location)	1 2 3 X	1 2 3 X
Being able to delete all traces of my web content	1 2 3 X	1 2 3 X
Being connected to people on platforms like Twitter, Github, Facebook etc.	1 2 3 X	1 2 3 X
Having my web content reach a large audience	1 2 3 X	1 2 3 X
Having my web content reach a specific audience	1 2 3 X	1 2 3 X
Discovering web content posted by people with similar interests as me (e.g. recommendation systems, shared posts by friends, searching by hashtags)	1 2 3 X	1 2 3 X
Preventing web content from being censored (by governments or platforms etc.)	1 2 3 X	1 2 3 X
Protecting people from harassment	1 2 3 X	1 2 3 X
Decreasing spread of misinformation	1 2 3 X	1 2 3 X
Making the web accessible to people with disabilities	1 2 3 X	1 2 3 X
<i>Making the web accessible to people with low technical skills.</i>	1 2 3 X	1 2 3 X
<i>Making the web accessible to people suffering under poverty</i>	1 2 3 X	1 2 3 X

Others?

- Especially if there are goals related to your values or ethical beliefs

Note: The categories in this list will be refined as early data analysis identifies and clarifies value-driven goals that are important to the IndieWeb.

If there are any differences in scores between columns (i.e. the interviewee indicates that they care about an issue more or less than the IndieWeb community at large), then ask the following questions about those items:

- Why do you think the IndieWeb community cares more/less about this issue than you do?
- Does this difference of opinion affect the types of work you do related to IndieWeb? If so, how?

Outcomes:

- 18) Of the goals you just identified as important to you:
- a) Which have you been successful in achieving through your technical work? What technical decisions support those goals?
 - b) Which have been unsuccessful. Why?
 - c) Have you ever had to compromise these goals to maintain compatibility with third-party platforms or other systems?

Demographic questions

I'd like to ask some demographic questions if that's okay. Again, if you prefer not to answer any of these questions that'd totally fine.

- 19) What is the highest level of education you've completed?
- 20) How do you identify your race?
- 21) How do identify your gender?
- 22) How old are you? (You can be approximate if you prefer – preferably at least to the decade)
- 23) In what city and country do you live?

Concluding questions

- 24) Is there anyone that you know of personally that you would recommend as a participant for this study, or someone that could offer a different perspective? Please feel free to pass on my information to them and have them contact me at jack.jamieson@mail.utoronto.ca.
- 25) Is there anything I didn't ask that you think is important?

Appendix D

Procedure for identifying IndieWeb-related repositories on GitHub

1. Search IndieWeb’s wiki for every instance of the text string ‘github.com/’ and record the full link (e.g. <https://github.com/indieweb/wiki> or <https://github.com/jackjamieson2/yarns-microsub-server>). Where a link referred to a specific object inside a repo, such as an issue thread or commit, the repo link was extracted. (e.g. the repo <https://github.com/snarfed/bridgy> was identified in the link <https://github.com/snarfed/bridgy/issues/42>). This identified 1,224 unique results.
2. Query each link through GitHub’s API and check if it is a valid repository. Of the links identified in the previous step, 1,006 were verified as referring to valid repos. The invalid links may have contained typos, been deleted, or no longer public.
3. Only keep repos whose name, description, or readme file contain any of the following IndieWeb-related keywords: *indieweb*, *indie web*, *microformats 2*, *microformats2*, *mf2*, *jf2*, *webmention*, *indieauth*, *micropub*, *microsub*, *websub*, *pubsubhubbub*, *withknob*, *idno*, *micro.blog*, *bridgy*. These keywords are related to IndieWeb standards, as well as a select number of individual IndieWeb applications. This removes projects that may have been referred to in IndieWeb’s wiki, but which are not IndieWeb-focused. 413 repos were found to contain these keywords.
4. Since there is a high likelihood that the owner of one IndieWeb-related repo also works on additional IndieWeb projects, for each repo owner, add any of their other repos that contain IndieWeb keywords as described in step 3. This increases the total repo list to 1,004 items.
5. Add repos containing commits (changes to code) that reference any the IndieWeb-related repos already identified. A typical case for this to occur is when repos A and B interoperate in some way, and a bug discovered in A is resolved through a change to B. For example, B may be a code library used by A, or the two repos may be two applications with a client/server relationship. This increased the size of the repo list to 1,139.

6. An additional round of filtering removed any newly discovered items from step 5 that did not contain keywords as defined in step 3, and removed any duplicates that were not identified before. This reduced the number of repositories to 1,079.
7. A final pass fetched all remaining repositories from the *indieweb*, *microformats*, and *idno* organizations on GitHub. These organizations are devoted to IndieWeb or IndieWeb-related projects, so even repositories that were not captured by the previous steps are relevant for this research. This increased the repository count to 1,102.

Appendix E

Keywords used to validate LDA topic models

This appendix presents a list of keywords that were used as one step in validating topic models generated using LDA. The distribution of these keywords was compared across documents in each topic, with an expectation that keywords I knew to be semantically related would be generally clustered together in the LDA model.

E.1 Principles (strict list)

- principle
- principles

E.2 Principles (fuzzy list)

This list included terms related to principles, selected using the procedure explained in Appendix F.

- principles
- principle
- own your data
- /own-your-data
- publish visible data
- humans first
- /DRY
- make what you need
- build tools for yourself

- /scratch_your_own_itch
- scratch your own itch
- /posse
- use what you make
- selfdogfood
- dogfood
- selfdogfooding
- document your stuff
- open source your stuff
- /ux
- /design
- /UX
- ux first
- /design,
- /modularity
- platform agnostic platforms
- longevity
- <https://adactio.com/articles/1522/>
- plurality
- /monoculture
- /fun
- /code-of-conduct
- code of conduct
- /why
- <http://www.wired.com/2013/12/indieweb-principles/>
- /friendly
- <https://web.archive.org/web/20190405080103/http://anouska.net/2018/02/blogging-and-why-its-important-to-bring-it-back/>
- https://amp.fastcompany.com/90202172/why-bad-technology-dominates-our-lives-according-to-don-norman?_indieweb_impression=true

- <https://justinjackson.ca/webmaster/>
- <https://twitter.com/kevinmarks/status/983848986099535874>
- <https://twitter.com/Shoq/status/983827715370422272>
- <https://twitter.com/pfrazee/status/1002827410621034496>
- <https://twitter.com/sonniesedge/status/1160840361079709696>

E.3 IndieWeb related standards

- webmention
- micropub
- websub
- jf2
- indieauth
- vouch
- salmention
- microsub
- microformats
- spec
- standard

E.4 IndieWeb events

- homebrew website club
- hwc
- indiewebcamp
- indieweb camp
- indie web camp
- iwc
- indieweb summit
- indie web summit
- iws

E.5 IndieWeb-related applications/software

Most of these items are copied from <https://indieweb.org/projects>. Some items not listed at that URL are included because they refer to well-known or prominent applications. App names that were common words were excluded if there was a reasonable expectation that they would predominantly be used in chat to refer to subjects other than the IndieWeb project with that name.

(Two sets of keywords were originally tested: (1) A *strict* version that only drew from <https://indieweb.org/projects> and (2) a *fuzzy* version including additional items. The results were not appreciably different so only the fuzzy version was retained and used.)

- withknown
- known
- idno
- bridgy
- webmention.io
- dobrado
- falcon
- franciscms
- kaku
- p3k
- wwwtech
- publ
- abode
- ferocity
- gopost
- hakkan
- dark matter
- pump.io
- sadlittlewebjournal
- taproot
- transformative
- postly
- voto

- neonblog
- triki
- indiereader
- indieweb reply
- own your comments
- phubb
- pingback2hook
- stapibas
- whisperfollow
- pushl
- authl
- quill
- alltogethernow
- yarns
- monocle
- alltogethernow
- quill
- yarns
- indiebookclub
- inkstone
- grumble
- kapowski
- micropublish
- micropub-xmpp-bot
- omnibear
- ownyourcheckin
- ownyourgram
- ownyourswarm
- photopotr

- pushupcounter
- screech
- slater
- sunlit
- micro.blog
- teacup
- woodwind
- aperture

Appendix F

Procedure for selecting 'Principle-related' keywords

Keywords listed in this section were derived from an article on IndieWeb's wiki about "principles for building on the IndieWeb" (IndieWeb.org 2020i). The following procedure was used to generate this list.

1. Start the list with two items: "principle" and "principles."
2. Identify all outbound links from <https://indieweb.org/principles>
3. For each link, add the link's text and URL to the list of principles. Some links were excluded if their use in IndieWeb's chat is unlikely to refer to a principle specifically (e.g. a link to http://en.wikipedia.org/wiki/Proof-of-work_system is presented as supporting material to one of IndieWeb's principles, but does not itself refer to a principle).
4. Add principle headings if they differ from link text and are meaningful. Similarly key phrases from within the description of each principles are added if they are suitable as summations that principle and therefore may be used in IndieWeb's chat to that effect. For example, the phrase 'ux first' is included as a commonly used summation of principles #7: "UX and design is more important than protocols, formats, data models, schema etc."
5. Add keywords from the opening paragraph of the *principles* article, which presents a more concise version of the list of principles.
6. Exclude keywords that are ambiguous or broad and therefore likely to be used to refer to concepts beyond IndieWeb's principles. For example, "document" and "open source" are not included because most instances of these terms in IndieWeb's chat are not related to discussions of IndieWeb's principles.

Appendix G

Timeline of IndieWeb events by location

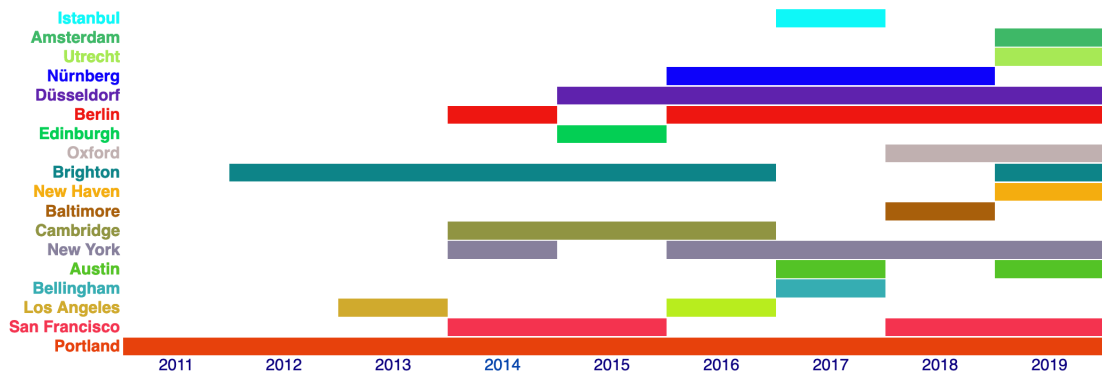


Figure G.1: Timeline of IndieWeb events by location. Reproduced from <https://indieweb.org/IndieWebCamps> as of May 24, 2020.

Appendix H

Top terms for 16 topic model

Table H.1: Top terms for 16 topic model.
Generated from pyLDAVis with $\lambda = 0.6$

#	30 most relevant terms
1	list, read, follow, long, medium, thing, people, idea, podcast, track, listen, lot, sound, social, interesting, time, site, good,story, news, thought, sound_like, superfeedr, websub, similar, create, audience, social_medium, reading, page
2	good, morning, week, event, hwc, good_morning, iwc, free, indiewebcamp, night, weekend, tom-morris, feel.free, feel, join, plan, venue, evening, happy, tomorrow, indieweb, bear, sf, tonight, nyc, remote, berlin, summit, meet, hope
3	irc, loqi, app, search, spam, slack, block, mobile, answer, android, high, log, bot, pingback, bad, vouch, question, native, low, io, good, engine, bridge, true, lol, channel, offline, slow, priority, stop
4	micropub, server, endpoint, client, indieauth, token, log, user, sign, domain, request, redirect, login, key, auth, quill, support, https, error, access, microsub, micropub_endpoint, verify, scope, provider, oauth, config, require, openid, aperture
5	web, indieweb, open, website, project, build, personal, community, people, site, bret, host, tool, indie, service, mastodon, open_source, pay, platform, software, goal, domain, tech, social, source, focus, blog, building, cms, personal_site

- 6 google, email, people, private, public, account, number, phone, notification, channel, archive, group, friend, push, dev, internet, contact, message, datum, worth, thing, chat, talk, apple, export, drive, lot, backup, org, device
-
- 7 browser, css, icon, work, style, js, image, text, yeah, screen, load, bit, space, firefox, chrome, extension, play, table, small, logo, size, weird, background, alt, svg, book, ha, font, tab, view
-
- 8 plugin, wordpress, issue, fix, change, theme, wp, kind, work, add, update, version, post_kind, release, option, pr, default, custom, semantic, core, pull, remove, merge, setting, support, break, feature, code, syndication, linkback
-
- 9 webmention, url, send, link, publish, mention, check, location, bridgy, source, date, send_webmention, target, find, receive, ah, checkin, map, ping, hmm, update, yeah, guess, handle, timezone, hm, accept, bridgy_publish, slug, wm
-
- 10 file, html, php, code, test, json, datum, store, write, form, library, spec, return, header, parser, database, python, language, string, line, storage, output, script, error, function, format, array, pass, xml, markdown
-
- 11 run, work, github, stuff, easy, static, site, set, pretty, cool, git, local, thing, setup, jekyll, yeah, host, repo, deploy, build, nice, import, push, install, start, super, static_site, manage, locally, package
-
- 12 feed, page, entry, card, property, microformat, rel, author, h_card, h_entry, markup, parse, class, reader, content, rss, element, home, h_feed, summary, atom, inside, home_page, cite, url, item, mark, object, imply, homepage
-
- 13 document, case, use_case, design, problem, specific, real, point, world, solve, term, exist, agree, implementation, provide, capture, practice, discussion, general, plain, ux, human, citation, documentation, opinion, argument, common, care, actual, simple
-
- 14 day, year, wiki, time, ago, edit, session, minute, video, haha, month, stream, demo, today, cool, watch, couple, live, room, awesome, lol, record, hour, remember, hear, audio, youtube, hey, oops, mediawiki
-
- 15 post, photo, type, tag, note, article, content, sense, make, blog, make_sense, person, bookmark, add, blog_post, image, category, text, flickr, title, instagram, kind, display, emoji, favorite, collection, create, field, fragmentation, hashtag
-
- 16 comment, reply, twitter, tweet, post, link, posse, copy, show, kylewm, twitt, fb, context, original, facebook, ui, delete, silo, bridgy, site, button, repost, syndicate, rsvp, permalink, thread, click, action, display, quote
-

Appendix I

Proportion of chat clusters related to each topic over time (observed).

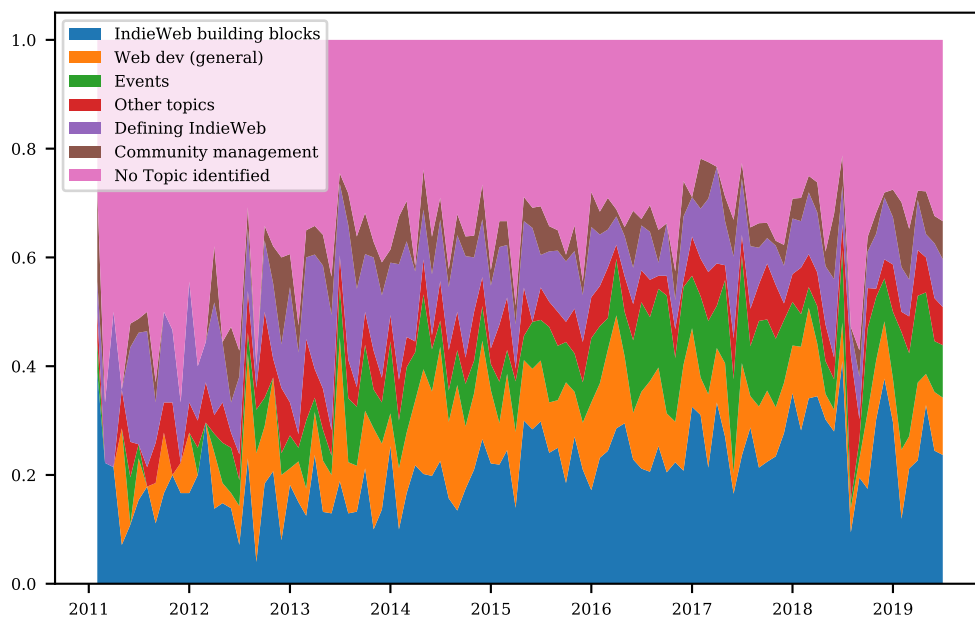


Figure I.1: Proportion of chat clusters related to each topic over time.

Figure I.1 presents a stackplot chart showing the proportion of observations associated with each topic over time. This figure charts the data as observed, by contrast with Figure 5.11 in Section 5.2.5, which charts the trend as calculated by the seasonal decomposition function in *statsmodels*.¹

¹See https://www.statsmodels.org/dev/generated/statsmodels.tsa.seasonal.seasonal_decompose.html

Appendix J

Network graphs of GitHub network by year

J.1 2011

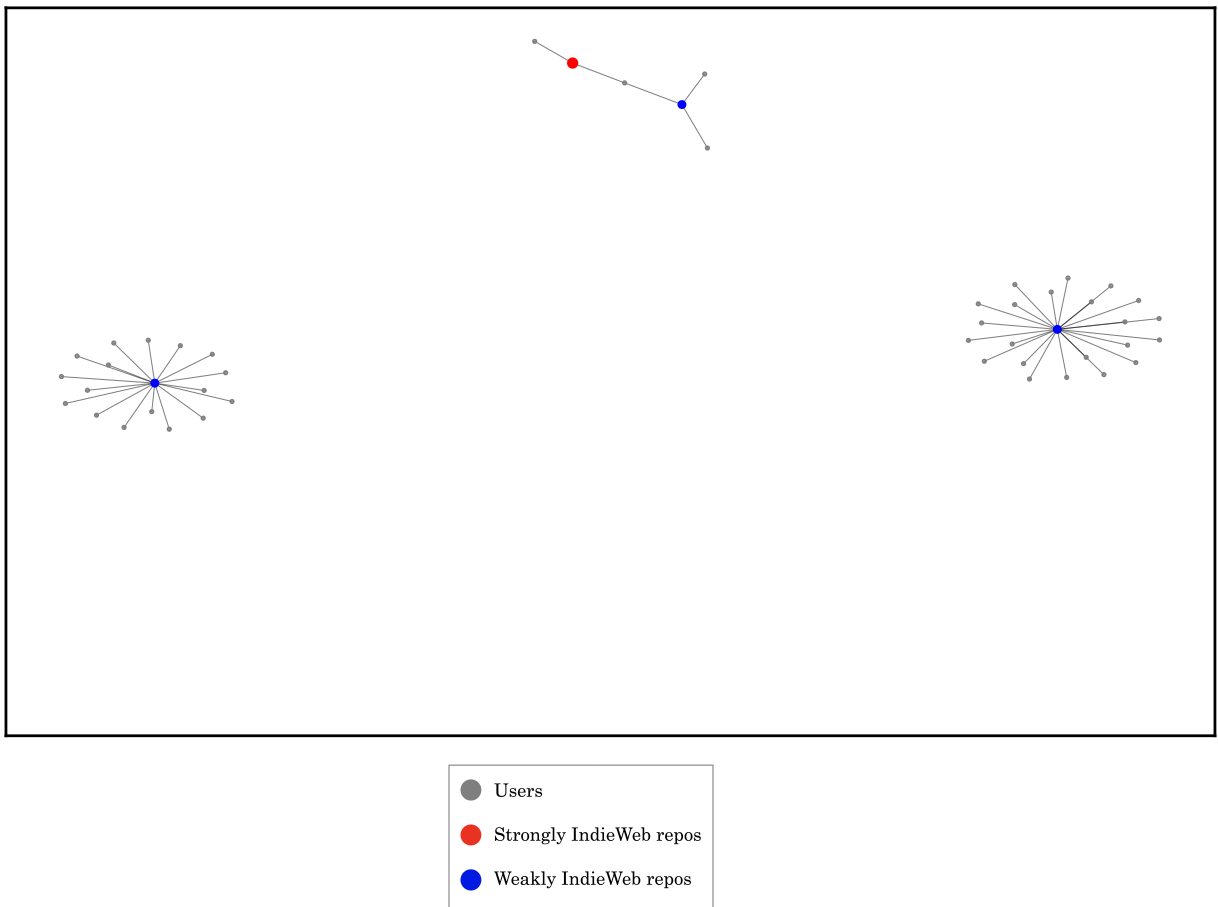


Figure J.1: Network chart of IndieWeb-related GitHub repos and users during 2011.

J.2 2012

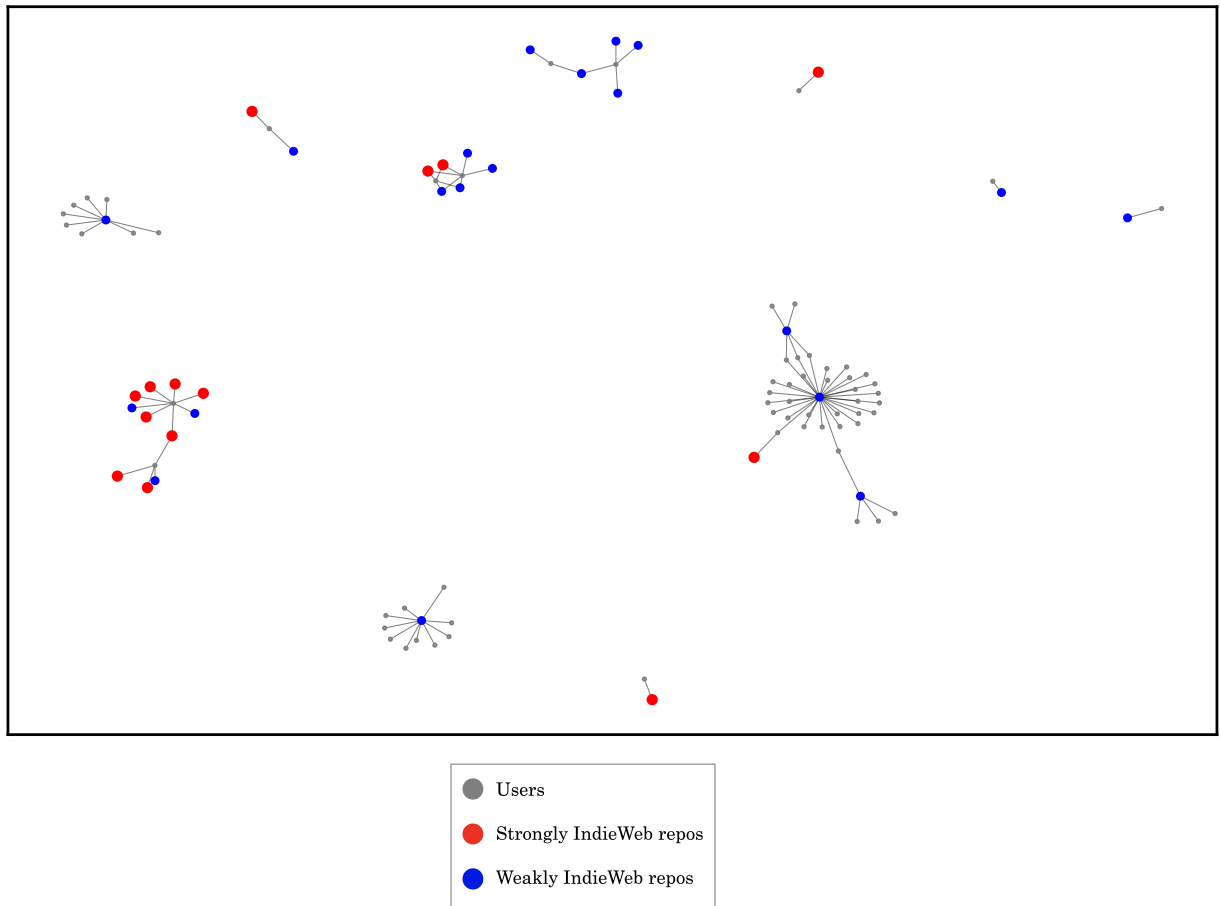


Figure J.2: Network chart of IndieWeb-related GitHub repos and users during 2012.

J.3 2013

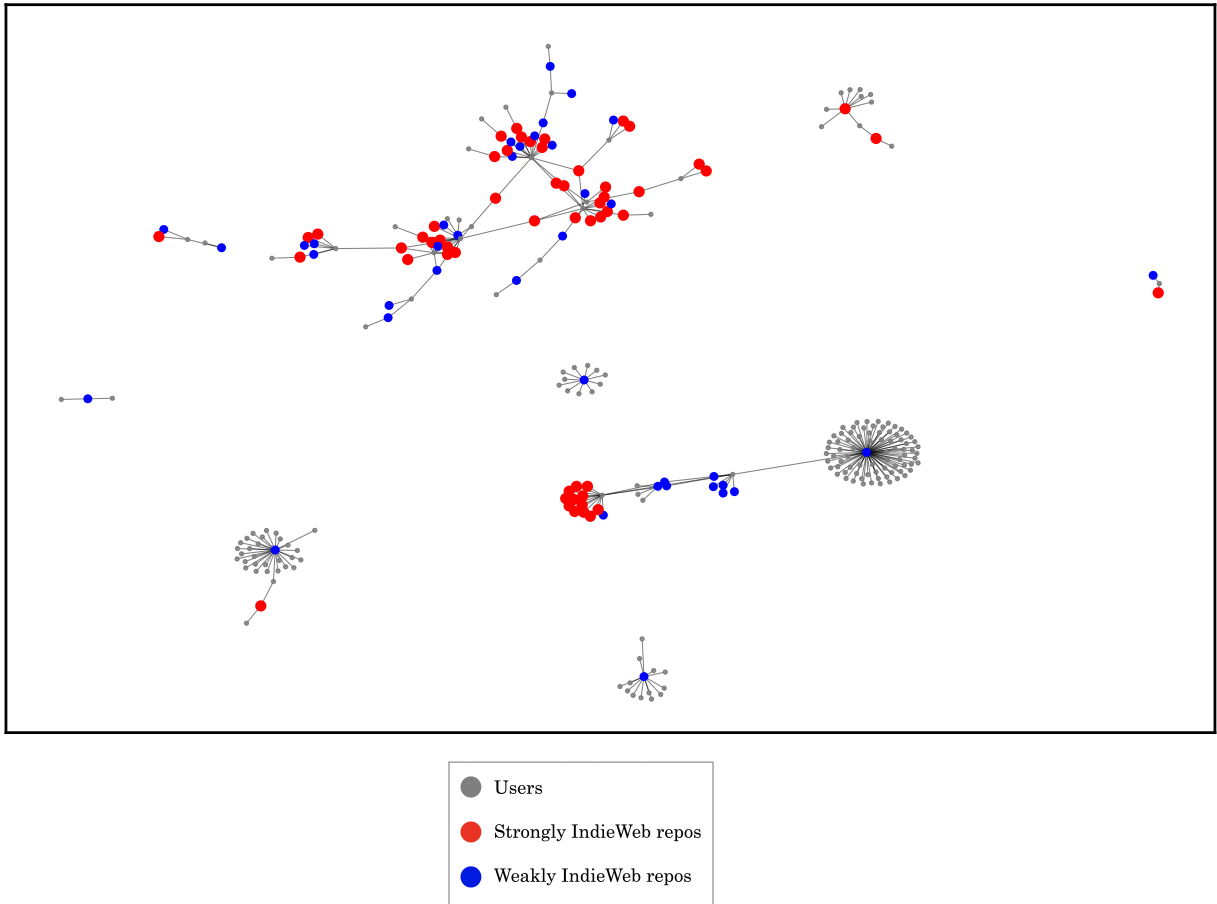


Figure J.3: Network chart of IndieWeb-related GitHub repos and users during 2013.

J.4 2014

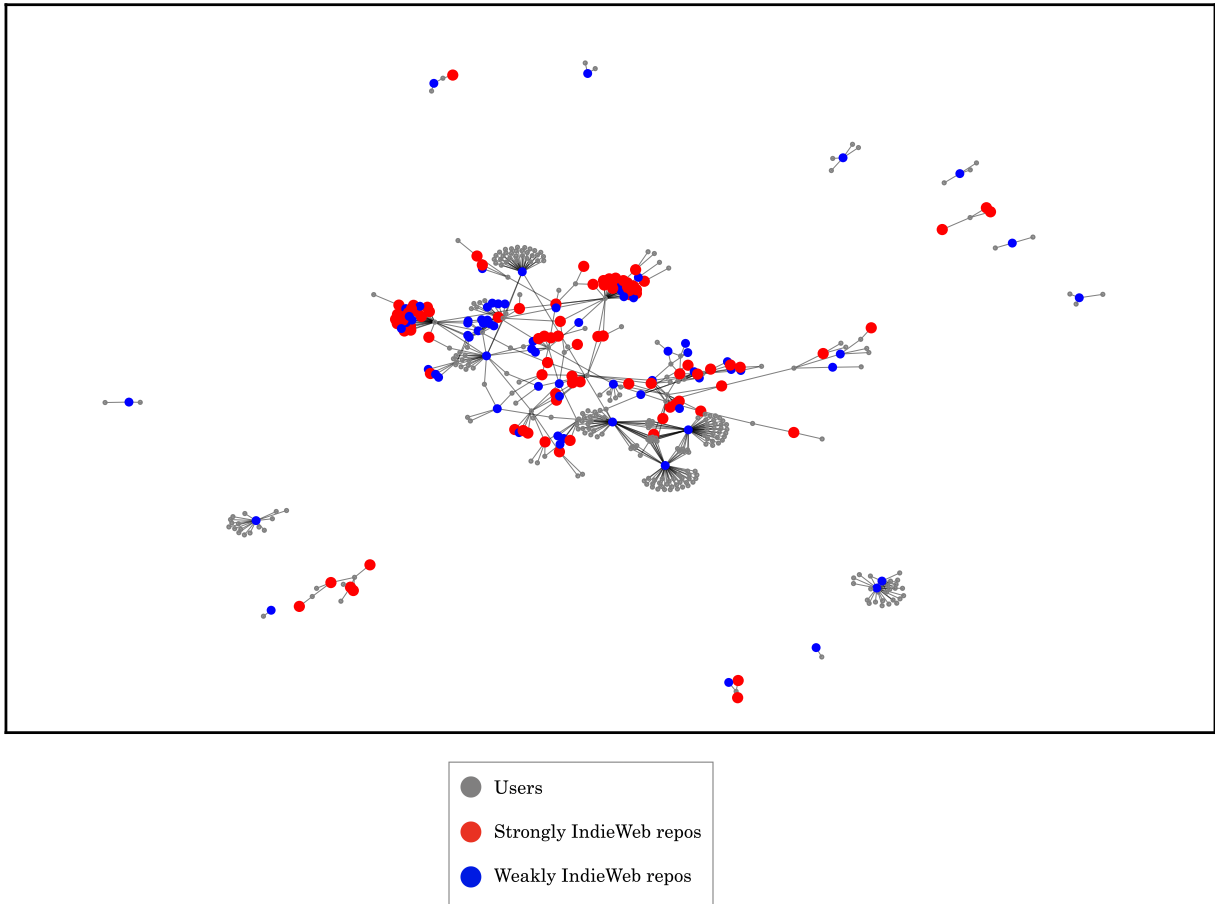


Figure J.4: Network chart of IndieWeb-related GitHub repos and users during 2014.

J.5 2015

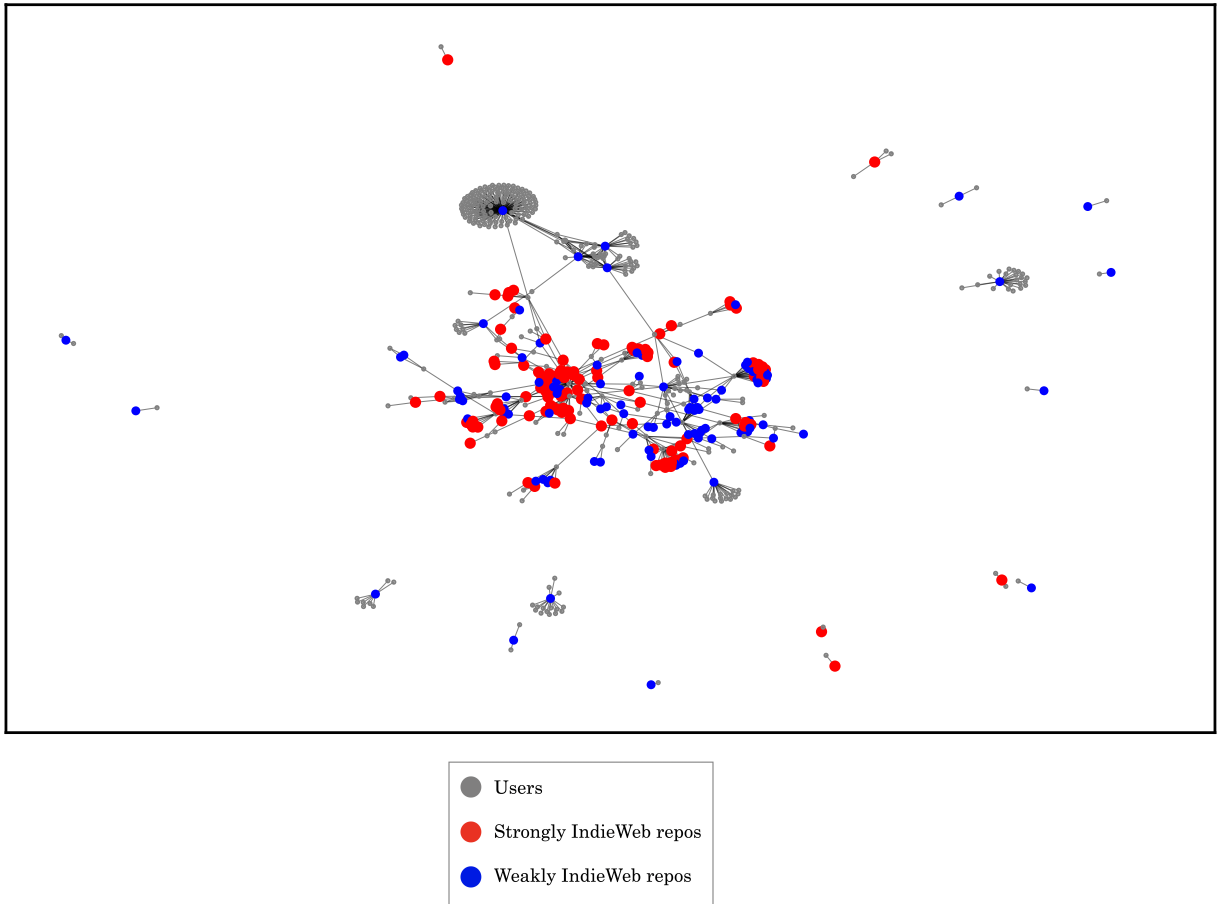


Figure J.5: Network chart of IndieWeb-related GitHub repos and users during 2015.

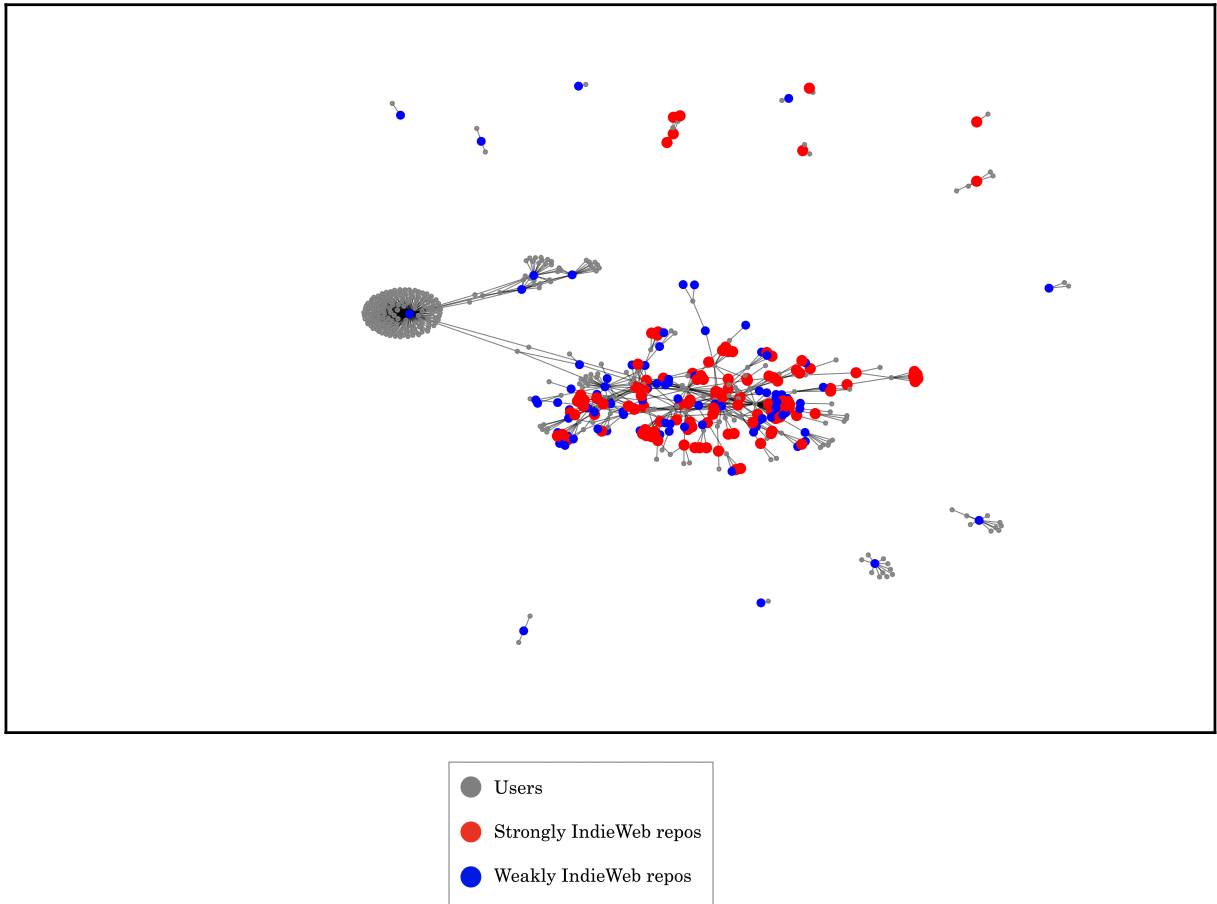
J.6 2016

Figure J.6: Network chart of IndieWeb-related GitHub repos and users during 2016.

J.7 2017

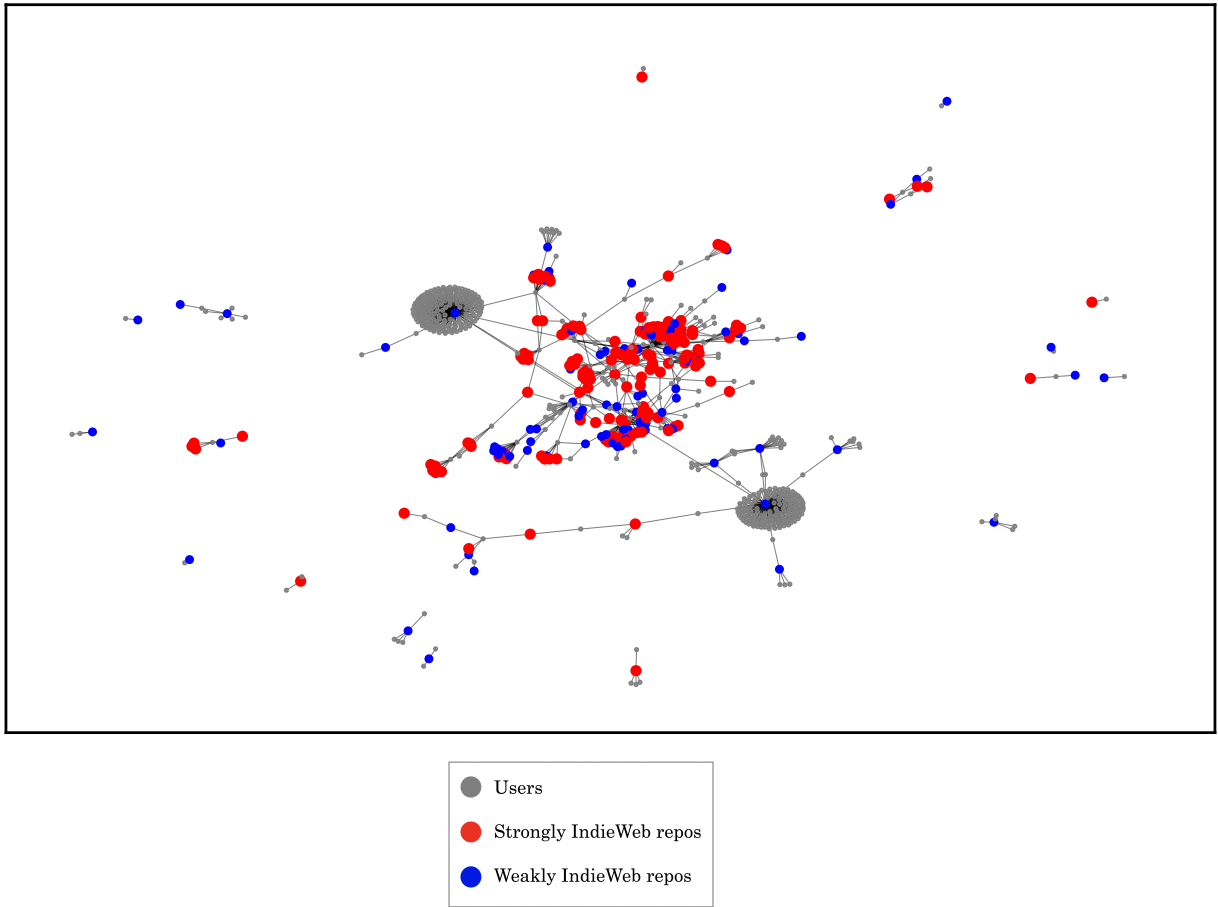


Figure J.7: Network chart of IndieWeb-related GitHub repos and users during 2017.

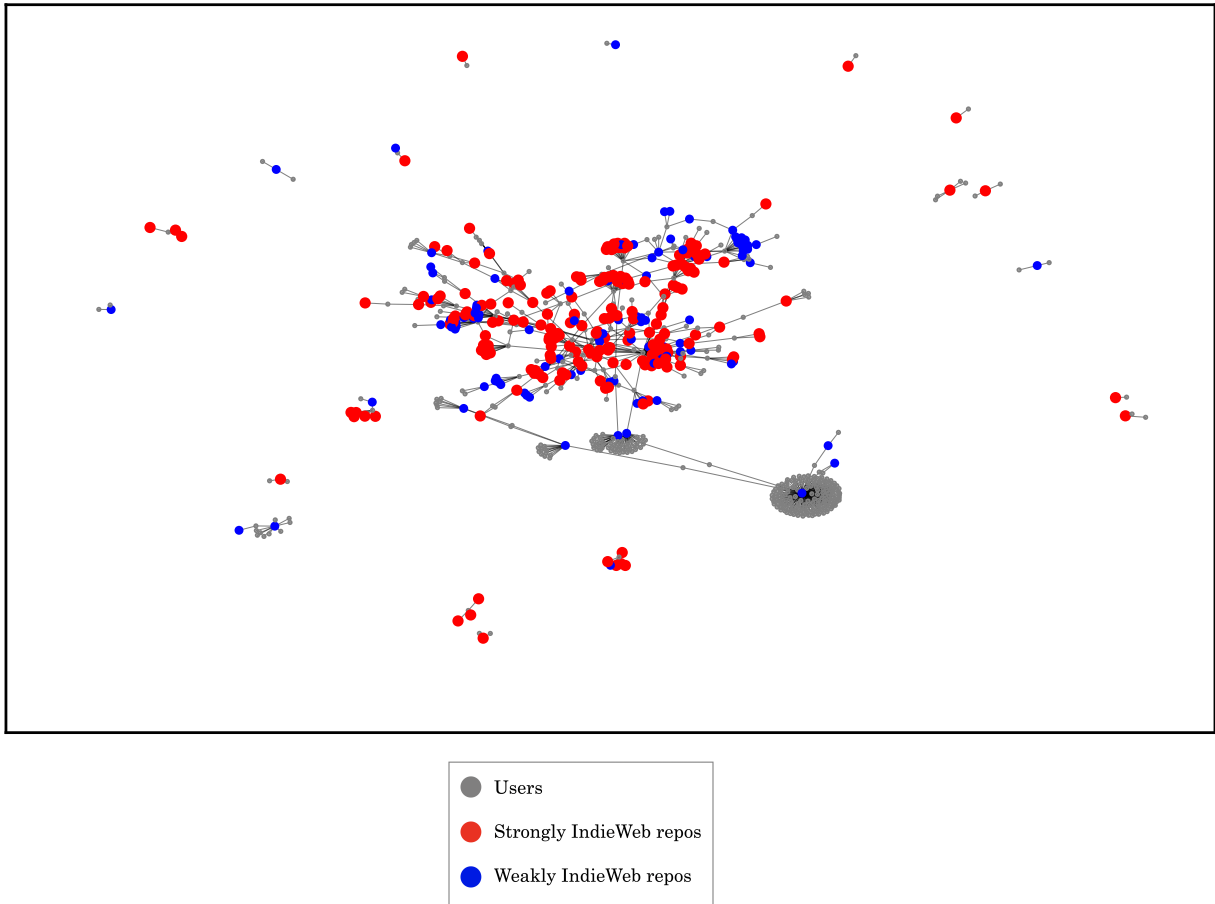
J.8 2018

Figure J.8: Network chart of IndieWeb-related GitHub repos and users during 2018.

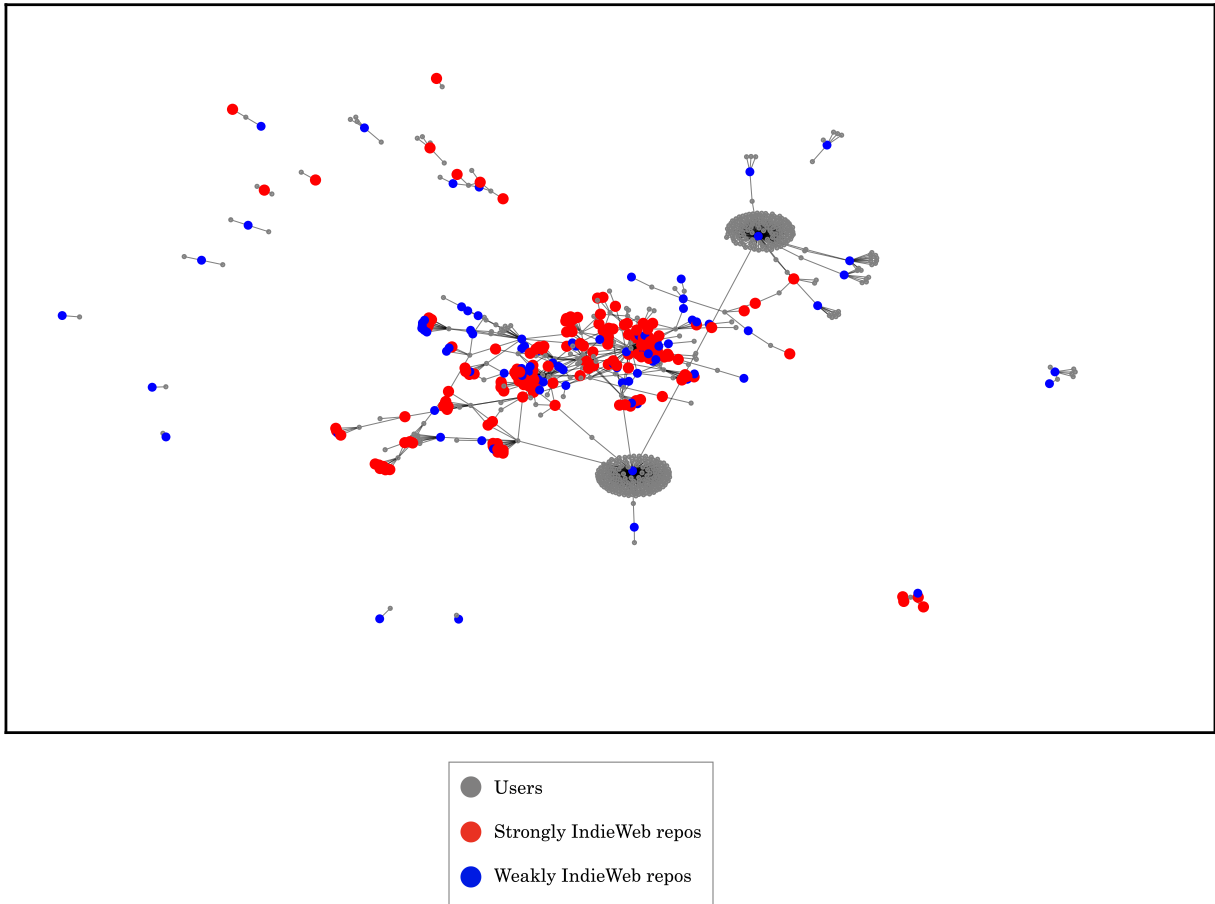
J.9 2019

Figure J.9: Network chart of IndieWeb-related GitHub repos and users from Jan 1, 2019 through May 17, 2019.

J.10 All-time

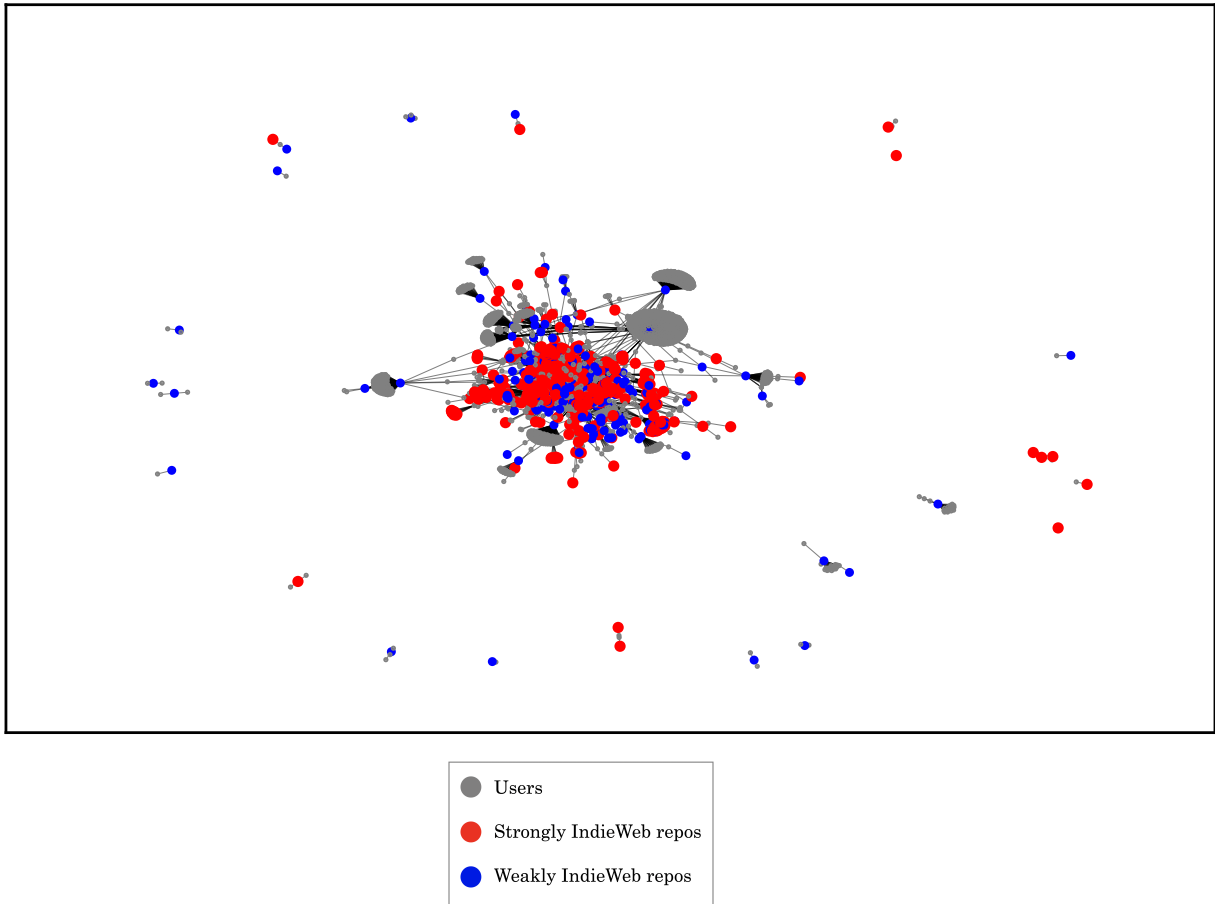


Figure J.10: Network chart of IndieWeb-related GitHub repos and users from Jan 1, 2011 through May 17, 2019.

Appendix K

Mann-Whitney results

K.1 GitHub repo degree centrality by location of IndieWeb keywords

Table K.1: Mann-Whitney U-test results: GitHub repo degree centrality by location of IndieWeb keywords.

IndieWeb keywords	Obs	Rank sum	Expected
In README only	285	111082	104025
In repo name or description	444	155003	266085
Combined	729	266085	266085
Unadjusted variance	7697850.00		
Adjustment for ties	-1.22e+06		
Adjusted variance	6474455.82		
Z	2.773		
Prob > z	0.0055		
Mann-Whitney U	56213		
Cliff's delta	-.11153785		

Bibliography

- Agre, Philip (1997a). *Computation and Human Experience*. Cambridge; New York: Cambridge University Press. ISBN: 978-0-521-38432-2 978-0-521-38603-6.
- (1997b). “Toward a Critical Technical Practice: Lessons Learned in Trying to Reform AI”. In: *Bridging the Great Divide: Social Science, Technical Systems, and Cooperative Work, Mahwah, NJ: Erlbaum*, pp. 131–157. URL: <http://polaris.gseis.ucla.edu/pagre/critical.html> (visited on 09/29/2015).
- Andreessen, Marc (Sept. 2007). *The Three Kinds of Platforms You Meet on the Internet*. URL: <https://web.archive.org/web/20071002031605/http://blog.pmarca.com/2007/09/the-three-kinds.html> (visited on 07/29/2016).
- Atton, Chris (2002). *Alternative Media*. en. London ; Thousand Oaks [Calif.]: SAGE. ISBN: 978-0-7619-6770-5 978-0-7619-6771-2.
- Baran, Paul (Aug. 1964). *On Distributed Communication Networks*. Tech. rep. RM-4230-PR. Santa Monica, CA: Rand Corporation.
- Barbrook, Richard and Andy Cameron (Jan. 1996). “The Californian Ideology”. en. In: *Science as Culture* 6.1, pp. 44–72. ISSN: 0950-5431, 1470-1189. DOI: 10.1080/09505439609526455. URL: <http://www.tandfonline.com/doi/abs/10.1080/09505439609526455> (visited on 09/30/2020).
- Bardzell, Jeffrey, Shaowen Bardzell, and Mark A. Blythe, eds. (2018). *Critical Theory and Interaction Design*. Cambridge, Massachusetts: The MIT Press. ISBN: 978-0-262-03798-3.
- Barrett, Ryan (2017). *Indie Map*. URL: <http://www.indiemap.org/> (visited on 11/02/2018).
- (Aug. 2018). *RIP Facebook for Bridgy*. en-US. URL: https://snarfed.org/2018-08-03_rip-facebook-for-bridgy (visited on 01/26/2019).
- (Jan. 2019). *(Personal Correspondence)*.
- Bell, Genevieve and Paul Dourish (Jan. 2007). “Yesterday’s Tomorrows: Notes on Ubiquitous Computing’s Dominant Vision”. en. In: *Personal and Ubiquitous Computing* 11.2, pp. 133–143. ISSN: 1617-4909, 1617-4917. DOI: 10.1007/s00779-006-0071-x. URL: <http://link.springer.com/10.1007/s00779-006-0071-x> (visited on 11/03/2015).
- Belshaw, Doug (Jan. 2020). *Microcast #081 – Anarchy, Federation, and the IndieWeb*. en-GB. URL: <https://thoughtshrapnel.com/2020/01/01/microcast-81/> (visited on 09/30/2020).
- Beninger, Kelsey (Jan. 2017). “Social Media Users’ Views on the Ethics of Social Media Research”. en. In: *The SAGE Handbook of Social Media Research Methods*. Ed. by Luke Sloan and Anabel Quan-Haase. London: SAGE, pp. 57–73. ISBN: 978-1-4739-8721-0.
- Berners-Lee, Tim (2010). “Long Live the Web”. In: *Scientific American* 303.6, pp. 80–85. URL: <http://www.nature.com/scientificamerican/journal/v303/n6/full/scientificamerican1210-80.html> (visited on 10/11/2014).

- Berners-Lee, Tim (June 2016). *Re-Decentralizing the Web: Some Strategic Questions*. San Francisco, CA. URL: http://archive.org/details/DWebSummit2016_Keynote_Tim_Berners_Lee (visited on 08/28/2019).
- Berners-Lee, Tim and Daniel W. Connolly (Nov. 1995). *Hypertext Markup Language - 2.0*. URL: <https://tools.ietf.org/html/rfc1866> (visited on 01/11/2017).
- Berners-Lee, Tim and Mark Fischetti (1999). *Weaving the Web: The Original Design and Ultimate Destiny of the World Wide Web by Its Inventor*. English. New York: HarperCollins. ISBN: 0-06-251586-1 978-0-06-251586-5 0-06-251587-X 978-0-06-251587-2.
- Birkinbine, Benjamin J. (2015). "Conflict in the Commons: Towards a Political Economy of Corporate Involvement in Free and Open Source Software". In: *The Political Economy of Communication 2.2*, pp. 3–19.
- (Feb. 2020). *Incorporating the Digital Commons: Corporate Involvement in Free and Open Source Software*. en. University of Westminster Press. ISBN: 978-1-912656-42-4. DOI: 10.16997/book39. URL: <https://uwestminsterpress.co.uk/site/books/10.16997/book39/> (visited on 03/10/2020).
- Blei, David M. et al. (May 2003). "Latent Dirichlet Allocation". In: *Journal of Machine Learning Research* 3.4/5, pp. 993–1022. ISSN: 15324435.
- Blumer, Herbert (1954). "What Is Wrong with Social Theory?" In: *American Sociological Review* 18, pp. 3–10. URL: https://brocku.ca/MeadProject/Blumer/Blumer_1954.html (visited on 03/11/2020).
- Borgmann, A. (Jan. 2010). "Reality and Technology". en. In: *Cambridge Journal of Economics* 34.1, pp. 27–35. ISSN: 0309-166X, 1464-3545. DOI: 10.1093/cje/ben055. URL: <http://cje.oxfordjournals.org/cgi/doi/10.1093/cje/ben055> (visited on 10/05/2015).
- Bowker, Geoffrey, Susan Leigh Star, Les Gasser, et al. (2014). *Social Science, Technical Systems, and Cooperative Work: Beyond the Great Divide*. English. Hoboken: Taylor and Francis. ISBN: 978-1-317-77876-9. URL: <http://public.ebookcentral.proquest.com/choice/publicfullrecord.aspx?p=1689049> (visited on 04/12/2020).
- Bowker, Geoffrey, Susan Leigh Star, William Turner, et al. (1997). *Social Science, Technical Systems and Cooperative Work: Beyond The Great Divide*. USA: L. Erlbaum Associates Inc. ISBN: 978-0-8058-2403-2.
- Bowker, Geoffrey C., Karen Baker, et al. (2009). "Toward Information Infrastructure Studies: Ways of Knowing in a Networked Environment". en. In: *International Handbook of Internet Research*. Ed. by Jeremy Hunsinger, Lisbeth Klastrup, and Matthew Allen. Dordrecht: Springer Netherlands, pp. 97–117. ISBN: 978-1-4020-9788-1 978-1-4020-9789-8. URL: http://link.springer.com/10.1007/978-1-4020-9789-8_5 (visited on 02/23/2016).
- Bowker, Geoffrey C. and Susan Leigh Star (Aug. 2000). *Sorting Things out: Classification and Its Consequences*. English. Cambridge, Mass.: The MIT Press. ISBN: 978-0-262-52295-3.
- boyd, danah (June 2009). *The Not-So-Hidden Politics of Class Online*. New York, NY, USA. URL: <http://www.danah.org/papers/talks/PDF2009.html> (visited on 05/01/2020).
- (Apr. 2010). *Privacy and Publicity in the Context of Big Data*. URL: <http://www.danah.org/papers/talks/2010/WWW2010.html> (visited on 11/12/2019).
- (Sept. 2019). *Facing the Great Reckoning Head-On*. en. URL: <https://medium.com/@zephoria/facing-the-great-reckoning-head-on-8fe434e10630> (visited on 09/16/2019).

- Braverman, Harry (1998). *Labor and Monopoly Capital: The Degradation of Work in the Twentieth Century*. 25th anniversary ed. New York: Monthly Review Press. ISBN: 978-0-85345-940-8.
- Brey, Philip (Dec. 2000). “Disclosive Computer Ethics”. In: *SIGCAS Comput. Soc.* 30.4, pp. 10–16. ISSN: 0095-2737. DOI: 10.1145/572260.572264. URL: <http://doi.acm.org/10.1145/572260.572264> (visited on 01/18/2017).
- (2010). “Values in Technology and Disclosive Computer Ethics”. In: *The Cambridge Handbook of Information and Computer Ethics*. Ed. by L Floridi. Cambridge: Cambridge University Press, pp. 41–58.
- Bridgy (2020). *About - Bridgy*. URL: <https://bridgy/about> (visited on 05/01/2018).
- Brown, John Seely and Paul Duguid (1991). “Organizational Learning and Communities-of-Practice: Toward a Unified View of Working, Learning, and Innovation”. In: *Organization science* 2.1, pp. 40–57. URL: <http://pubsonline.informs.org/doi/abs/10.1287/orsc.2.1.40> (visited on 11/12/2015).
- Bruns, Axel (2008). *Blogs, Wikipedia, Second Life, and beyond: From Production to Producership*. en. Peter Lang. ISBN: 0-8204-8866-6.
- Brunton, Finn and Helen Nissenbaum (2015). *Obfuscation: A User’s Guide for Privacy and Protest*. eng. Technology/Politics. Cambridge, Mass.: MIT Press. ISBN: 978-0-262-33131-9 978-0-262-02973-5.
- Burkell, Jacquelyn and Priscilla Regan (July 2019). “Examining Expression in the Virtual Public: The Ethics of Harvesting Online Discussions for Research Purposes”. In: *10th International Conference on Social Media & Society*. Toronto.
- Carr, Nicholas (Dec. 2006). *Digital Sharecropping*. URL: <http://www.routhtype.com/?p=634> (visited on 12/01/2016).
- Çelik, Tantek (June 2014). *Why We Need the IndieWeb*. URL: <https://www.youtube.com/watch?v=HNmK07Gr4TE> (visited on 12/14/2016).
- (June 2016). *State of the IndieWeb*. Portland, OR. URL: https://www.youtube.com/watch?v=7zTolqW_I2g (visited on 02/02/2017).
- (June 2018). *State of the IndieWeb - IndieWeb Summit 2018*. Portland, OR. URL: <https://www.youtube.com/watch?v=Vcn9FJCEI1U%5C&list=PLk3TtIJ31hqodLN698UsRXsOX-WqLneJ6%5C&index=2> (visited on 04/28/2020).
- (Jan. 2020a). *Indieweb-Community-Inclusive-Anarchistic-Punk*. URL: <https://tantek.com/2020/010/t2/indieweb-community-inclusive-anarchistic-punk> (visited on 09/30/2020).
- (Jan. 2020b). *Indieweb-Community-Individual-Agency*. URL: <https://tantek.com/2020/010/t3/indieweb-community-individual-agency> (visited on 09/30/2020).
- Chang, Jonathan et al. (n.d.). “Reading Tea Leaves: How Humans Interpret Topic Models”. en. In: (), p. 10.
- Charmaz, Kathy (2006). *Constructing Grounded Theory*. en. London ; Thousand Oaks, Calif: Sage Publications. ISBN: 978-0-7619-7352-2 978-0-7619-7353-9.
- Cheney-Lippold, J. (Nov. 2011). “A New Algorithmic Identity: Soft Biopolitics and the Modulation of Control”. en. In: *Theory, Culture & Society* 28.6, pp. 164–181. ISSN: 0263-2764, 1460-3616. DOI: 10.1177/0263276411424420. URL: <http://tcs.sagepub.com/cgi/doi/10.1177/0263276411424420> (visited on 11/29/2016).
- Cheng, Jinghui and Jin L. C. Guo (Mar. 2019). “Activity-Based Analysis of Open Source Software Contributors: Roles and Dynamics”. In: *arXiv:1903.05277 [cs]*. arXiv: 1903.05277 [cs]. URL: <http://arxiv.org/abs/1903.05277> (visited on 05/27/2019).

- Chernova, Yuliya (Aug. 2017). "In Landmark, Filecoin Offering Hauls in \$52 Million; Funding Said to Be One of the Nation's First Initial Coin Offerings in Regulatory Compliance - U.S. Newsstream - ProQuest". en. In: *Wall Street Journal*. URL: <https://www.proquest.com/docview/1925927851> (visited on 07/08/2020).
- Cohen, Nicole S. (2012). "From Alienation to Autonomy: The Labour of Alternative Media". In: *Alternative Media in Canada*. Ed. by Kirsten Kozolanka, Patricia Mazepa, and David Skinner. Vancouver; Toronto: University of British Columbia Press.
- (Feb. 2013). "Commodifying Free Labor Online: Social Media, Audiences, and Advertising". en. In: *The Routledge Companion to Advertising and Promotional Culture*. Ed. by Matthew P. McAllister and Emily West. Routledge, pp. 177–191. ISBN: 978-1-135-09556-7.
- Coleman, E. Gabriella (Oct. 2010). "Ethnographic Approaches to Digital Media". en. In: *Annual Review of Anthropology* 39.1, pp. 487–505. ISSN: 0084-6570, 1545-4290. DOI: 10.1146/annurev.anthro.012809.104945. URL: <http://www.annualreviews.org/doi/10.1146/annurev.anthro.012809.104945> (visited on 11/07/2016).
- Coleman, Gabriella (2004). "The Political Agnosticism of Free and Open Source Software and the Inadvertent Politics of Contrast". en. In: *Anthropological Quarterly* 77.3, pp. 507–519. ISSN: 1534-1518. DOI: 10.1353/anq.2004.0035. URL: http://muse.jhu.edu/content/crossref/journals/anthropological_quarterly/v077/77.3coleman.html (visited on 12/01/2014).
- Cox, Andrew (Dec. 2005). "What Are Communities of Practice? A Comparative Review of Four Seminal Works". en. In: *Journal of Information Science* 31.6, pp. 527–540. ISSN: 0165-5515, 1741-6485. DOI: 10.1177/0165551505057016. URL: <http://journals.sagepub.com/doi/10.1177/0165551505057016> (visited on 02/24/2020).
- Crawley, Fred (Aug. 2013). *The US Government Is Taking Your Facebook Data. But It Doesn't Stop There*. en. URL: <https://www.newstatesman.com/business/2013/08/us-government-taking-your-facebook-data-it-doesnt-stop-there> (visited on 11/13/2019).
- Csikszentmihalyi, Mihaly (2009). *Flow: The Psychology of Optimal Experience*. eng. Nachdr. Harper Perennial Modern Classics. New York: Harper [and] Row. ISBN: 978-0-06-133920-2.
- Damarin, Amanda K. (Jan. 2013). "The Network-Organized Labor Process: Control and Autonomy in Web Production Work". en. In: *Networks, Work and Inequality*. Ed. by Steve McDonald. Vol. 24. Emerald Group Publishing Limited, pp. 177–205. ISBN: 978-1-78190-539-5 978-1-78190-540-1. URL: <http://www.emeraldinsight.com/doi/abs/10.1108/S0277-2833%5C%282013%5C%290000024010> (visited on 09/22/2015).
- Data Analytics Group, University of Zurich (2019). *Pathpy*. URL: <https://www.pathpy.net/> (visited on 03/05/2020).
- Davies, Bethan (2005). "Communities of Practice: Legitimacy Not Choice". en. In: *Journal of Sociolinguistics* 9.4, pp. 557–581. ISSN: 1467-9841. DOI: 10.1111/j.1360-6441.2005.00306.x. URL: <http://onlinelibrary.wiley.com/doi/abs/10.1111/j.1360-6441.2005.00306.x> (visited on 06/07/2020).
- Denzin, Norman K. (2015). "Triangulation". en. In: *The Blackwell Encyclopedia of Sociology*. American Cancer Society. ISBN: 978-1-4051-6551-8. DOI: 10.1002/9781405165518.wbeost050.pub2. URL: <http://onlinelibrary.wiley.com/doi/abs/10.1002/9781405165518.wbeost050.pub2> (visited on 06/30/2020).

- Dewey, John (Jan. 1891). “Moral Theory and Practice”. eng. In: *International Journal of Ethics* 1. URL: <http://archive.org/details/jstor-2375407> (visited on 02/13/2020).
- (1927). “Search for the Great Community”. In: *The Public and Its Problems*. New York: Henry Holt and Company, pp. 148–184.
- Dijck, José van, Thomas Poell, and Martijn de Waal (2018). *The Platform Society: Public Values in a Connective World*. en. Oxford University Press. ISBN: 978-0-19-088977-7.
- Dourish, Paul (Dec. 2019). “User Experience as a Legitimacy Trap”. In: *Interactions* XXVI.6. URL: <http://interactions.acm.org/archive/view/november-december-2019/user-experience-as-legitimacy-trap> (visited on 06/28/2020).
- Downing, John (2001). *Radical Media: Rebellious Communication and Social Movements*. en. Thousand Oaks, Calif: Sage Publications. ISBN: 978-0-8039-5698-8 978-0-8039-5699-5.
- Dunbar-Hester, Christina (2014). “Radical Inclusion? Locating Accountability in Technical DIY”. In: *DIY Citizenship: Critical Making and Social Media*. Ed. by Matt Ratto and Megan Boler. Cambridge Massachusetts: The MIT Press. ISBN: 978-0-262-02681-9 978-0-262-52552-7.
- Dwyer, Jim (May 2010). “Four Nerds and a Cry to Arms Against Facebook”. en-US. In: *The New York Times*. ISSN: 0362-4331. URL: <https://www.nytimes.com/2010/05/12/nyregion/12about.html> (visited on 03/27/2020).
- Egelman, Serge (Apr. 2013). “My Profile Is My Password, Verify Me! The Privacy/Convenience Tradeoff of Facebook Connect”. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '13. Paris, France: Association for Computing Machinery, pp. 2369–2378. ISBN: 978-1-4503-1899-0. DOI: 10.1145/2470654.2481328. URL: <http://doi.org/10.1145/2470654.2481328> (visited on 06/17/2020).
- Ellul, Jacques (1964). *The Technological Society*. New York, NY, USA: Vintage.
- ESOMAR (2011). *ESOMAR Guidelines on Social Media Research*. URL: <https://www.esomar.org/uploads/public/knowledge-and-standards/codes-and-guidelines/ESOMAR-Guideline-on-Social-Media-Research.pdf> (visited on 02/04/2020).
- Espinha, Tiago, Andy Zaidman, and Hans-Gerhard Gross (Feb. 2014). “Web API Growing Pains: Stories from Client Developers and Their Code”. en. In: *Software Maintenance, Reengineering and Reverse Engineering (CSMR-WCRE), 2014*. Antwerp, Belgium: IEEE, pp. 84–93. ISBN: 978-1-4799-3752-3. DOI: 10.1109/CSMR-WCRE.2014.6747228. URL: <http://ieeexplore.ieee.org/document/6747228/> (visited on 05/07/2018).
- Evans, David S., Andrei Hagiu, and Richard Schmalensee (2006). *Invisible Engines: How Software Platforms Drive Innovation and Transform Industries*. Cambridge, Mass: MIT Press. ISBN: 978-0-262-05085-2.
- Facebook for Developers (2019a). *Post Not Returning with Correct Privacy*. en. URL: <https://developers.facebook.com/bugs/983622558349347/> (visited on 01/29/2019).
- (2019b). *Unusually Formatted Comment Ids (with Colons) Returned by /Me/Posts*. en. URL: <https://developers.facebook.com/bugs/786903278061433/> (visited on 01/29/2019).
- Faulkner, Susan and Anne Mcclard (Oct. 2014). “Making Change: Can Ethnographic Research about Women Makers Change the Future of Computing?” en. In: *Ethnographic Praxis in Industry Conference Proceedings 2014.1*, pp. 187–198. ISSN: 1559890X. DOI: 10.1111/1559-8918.01026. URL: <http://doi.wiley.com/10.1111/1559-8918.01026> (visited on 05/27/2020).

- Faulkner, Wendy (Oct. 2000). "Dualisms, Hierarchies and Gender in Engineering". en. In: *Social Studies of Science* 30.5, pp. 759–792. ISSN: 0306-3127. DOI: 10.1177/030631200030005005. URL: <https://doi.org/10.1177/030631200030005005> (visited on 06/08/2020).
- Feenberg, Andrew (Jan. 2002). *Transforming Technology: A Critical Theory Revisited*. English. Rev Sub edition. New York, N.Y: Oxford University Press. ISBN: 978-0-19-514615-8.
- Filippova, Anna and Hichang Cho (Feb. 2015). "Mudslinging and Manners: Unpacking Conflict in Free and Open Source Software". In: *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing*. CSCW '15. Vancouver, BC, Canada: Association for Computing Machinery, pp. 1393–1403. ISBN: 978-1-4503-2922-4. DOI: 10.1145/2675133.2675254. URL: <http://doi.org/10.1145/2675133.2675254> (visited on 05/26/2020).
- Finley, Klint (Aug. 2013). "Meet the Hackers Who Want to Jailbreak the Internet". In: *Wired*. ISSN: 1059-1028. URL: <https://www.wired.com/2013/08/indie-web/> (visited on 11/02/2018).
- Flanagan, Mary, Daniel C. Howe, and Helen Nissenbaum (Apr. 2005). "Values at Play: Design Tradeoffs in Socially-Oriented Game Design". In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '05. Portland, Oregon, USA: Association for Computing Machinery, pp. 751–760. ISBN: 978-1-58113-998-3. DOI: 10.1145/1054972.1055076. URL: <http://doi.org/10.1145/1054972.1055076> (visited on 05/05/2020).
- Flichy, Patrice (2007). *Understanding Technological Innovation: A Socio-Technical Approach*. en. Northampton: Edward Elgar Publishing. ISBN: 1-84720-862-2.
- Fox, Stephen (2000). "Communities Of Practice, Foucault And Actor-Network Theory". en. In: *Journal of Management Studies* 37.6, pp. 853–868. ISSN: 1467-6486. DOI: 10.1111/1467-6486.00207. URL: <http://onlinelibrary.wiley.com/doi/abs/10.1111/1467-6486.00207> (visited on 02/24/2020).
- Franklin, Ursula M. (2004). *The Real World of Technology*. Rev. ed. CBC Massey Lectures Series. Toronto, Ont. : Berkeley, CA: House of Anansi Press ; Distributed in the United States by Publishers Group West. ISBN: 978-0-88784-636-6.
- Freeman, Linton C. (1977). "A Set of Measures of Centrality Based on Betweenness". In: *Sociometry* 40.1, pp. 35–41.
- Friedman, Batya, ed. (1997). *Human Values and the Design of Computer Technology*. CSLI Lecture Notes no. 72. New York: Cambridge University Press. ISBN: 978-1-57586-081-7 978-1-57586-080-0.
- Friedman, Batya and David Hendry (2019). *Value Sensitive Design: Shaping Technology with Moral Imagination*. Cambridge, Massachusetts: The MIT Press. ISBN: 978-0-262-03953-6.
- Friedman, Batya, Daniel C. Howe, and Edward Felten (2002). "Informed Consent in the Mozilla Browser: Implementing Value-Sensitive Design". In: *System Sciences, 2002. HICSS. Proceedings of the 35th Annual Hawaii International Conference On*. IEEE, 10–pp. URL: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=994366 (visited on 11/18/2015).
- Friedman, Batya, Peter Kahn, and Alan Borning (2002). "Value Sensitive Design: Theory and Methods". In: *University of Washington technical report*, pp. 02–12. URL: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.11.8020%5C&rep=rep1%5C&type=pdf> (visited on 01/18/2017).
- Friedman, Batya and Peter H. Kahn (2003). "Human Values, Ethics, and Design". In: *Handbook of Human-Computer Interaction*. Ed. by J Jacko and A Sears. Mahwah, NJ: Lawrence Erlbaum Associates, pp. 1177–1201.
- Friedman, Batya, Peter H. Kahn, and Alan Borning (2006). "Value Sensitive Design and Information Systems". In: *Human-Computer Interaction in Management Information Systems: Foundations*. Ed.

- by P Zhang and D Galetta. *Advances in Management Information Systems*. Armonk, New York: M.E.Sharpe, pp. 348–372. URL: https://www.researchgate.net/profile/Terrell_Bynum/publication/228050676_Milestones_in_the_History_of_Information_and_Computer_Ethics/links/00b7d53c89d05bfffef000000.pdf%5C#page=104 (visited on 01/25/2016).
- Friedman, Batya and Helen Nissenbaum (1997). “Bias in Computer Systems”. In: *Human Values and the Design of Computer Technology*. Ed. by Batya Friedman. CSLI Lecture Notes no. 72. New York: Cambridge University Press, pp. 21–40. ISBN: 978-1-57586-081-7 978-1-57586-080-0.
- Fuchs, Christian (2012). “Critique of the Political Economy of Web 2.0 Surveillance”. en. In: *Internet and Surveillance: The Challenges of Web 2.0 and Social Media*. Ed. by Christian Fuchs et al. New York: Routledge, pp. 31–70. ISBN: 978-1-136-65526-5.
- Gehl, Robert W. (Apr. 2015a). *Building a Better Twitter: A Study of the Twitter Alternatives GNU Social, Quitter, Rstat.Us, and Twister*. en. SSRN Scholarly Paper ID 2595247. Rochester, NY: Social Science Research Network. URL: <https://papers.ssrn.com/abstract=2595247> (visited on 08/08/2019).
- (2015b). “The Case for Alternative Social Media”. In: *Social Media + Society* 1.2, pp. 1–12. DOI: 10.1177/2056305115604338.
- (2018). “Alternative Social Media: From Critique to Code”. en. In: *The SAGE Handbook of Social Media*. London: SAGE Publications Ltd, pp. 330–350. ISBN: 978-1-4129-6229-2 978-1-4739-8406-6. DOI: 10.4135/9781473984066.n19. URL: <http://sk.sagepub.com/reference/the-sage-handbook-of-social-media//i2724.xml> (visited on 08/08/2019).
- Geiger, R Stuart and David Ribes (2011). “Trace Ethnography: Following Coordination through Documentary Practices”. en. In: *IEEE*, pp. 1–10. ISBN: 978-1-4244-9618-1. DOI: 10.1109/HICSS.2011.455. URL: <http://ieeexplore.ieee.org/document/5718606/> (visited on 06/12/2018).
- Gibson, James J. (1986). “The Theory of Affordances”. en. In: *The Ecological Approach to Visual Perception*. Hillsdale, New Jersey: Lawrence Erlbaum Associates, Publishers, pp. 127–143.
- Gillespie, Tarleton (Feb. 2010). “The Politics of ‘platforms’”. In: *New Media & Society* 12.3, pp. 347–364. ISSN: 1461-4448, 1461-7315. DOI: 10.1177/1461444809342738. URL: <http://nms.sagepub.com/cgi/doi/10.1177/1461444809342738> (visited on 02/01/2014).
- (2018). *Custodians of the Internet: Platforms, Content Moderation, and the Hidden Decisions That Shape Social Media*. New Haven: Yale University Press. ISBN: 978-0-300-17313-0.
- Gillmor, Dan (Aug. 2014a). *The New Editors of the Internet*. en-US. URL: <https://www.theatlantic.com/technology/archive/2014/08/the-new-editors-of-the-internet/378983/> (visited on 05/24/2020).
- (Apr. 2014b). “Welcome to the Indie Web Movement”. en-US. In: *Slate*. ISSN: 1091-2339. URL: http://www.slate.com/blogs/future_tense/2014/04/25/indiewebcamps_create_tools_for_a_new_internet.html (visited on 11/02/2018).
- (Apr. 2014c). *Why the Indie Web Movement Is so Important*. URL: <http://dangillmor.com/2014/04/25/indie-web-important/> (visited on 11/17/2016).
- GNU GPLv2 (1991). *GNU General Public License, Version 2*. en. URL: <https://www.gnu.org/licenses/old-licenses/gpl-2.0.html> (visited on 05/04/2020).
- Google (Jan. 2019). *Shutting down Google+ for Consumer (Personal) Accounts on April 2, 2019 - Google+ Help*. URL: <https://support.google.com/plus/answer/9195133> (visited on 06/18/2020).

- Grossman, Lev (Dec. 2006). “You — Yes, You— Are TIME’s Person of the Year”. In: *Time*. ISSN: 0040-718X. URL: <http://content.time.com/time/magazine/article/0,9171,1570810,00.html> (visited on 08/19/2015).
- Gruzd, Anatoliy et al. (2018). “The State of Social Media in Canada 2017”. en. In: *SSRN Electronic Journal*. ISSN: 1556-5068. DOI: 10.2139/ssrn.3158771. URL: <https://www.ssrn.com/abstract=3158771> (visited on 09/04/2019).
- Guy, Amy (2017). “The Presentation of Self on a Decentralised Web”. PhD thesis. Edinburgh: University of Edinburgh. URL: <https://rhiaro.github.io/thesis/> (visited on 02/04/2020).
- Haimson, Oliver L. and Anna Lauren Hoffmann (June 2016). “Constructing and Enforcing ”Authentic” Identity Online: Facebook, Real Names, and Non-Normative Identities”. en. In: *First Monday* 21.6. ISSN: 13960466. URL: <http://firstmonday.org/ojs/index.php/fm/article/view/6791> (visited on 09/26/2016).
- Halpin, Harry (Oct. 2018). “Decentralizing the Social Web: Can Blockchains Solve Ten Years of Standardization Failure of the Social Web?” en. In: *INSCI’2018- 5th International Conference ‘Internet Science’*. St. Petersburg, Russia, p. 16.
- Haraway, Donna (1988). “Situated Knowledges: The Science Question in Feminism and the Privilege of Partial Perspective”. en. In: *Feminist Studies* 14.3, pp. 575–599. ISSN: 00463663. DOI: 10.2307/3178066. URL: <https://www.jstor.org/stable/3178066?origin=crossref> (visited on 06/20/2020).
- Hassenzahl, Marc and Noam Tractinsky (2006). “User Experience - a Research Agenda”. In: *Behaviour & Information Technology* 25.2, pp. 91–97.
- Haythornthwaite, Caroline (2001). “Exploring Multiplexity: Social Network Structures in a Computer-Supported Distance Learning Class”. In: *The information society* 17.3, pp. 211–226. URL: <http://www.tandfonline.com/doi/abs/10.1080/01972240152493065> (visited on 10/22/2015).
- Helmond, Anne (2015). “The Web as Platform: Data Flows in Social Media”. PhD thesis. Amsterdam: University of Amsterdam.
- Hora, Andre et al. (Sept. 2015). “How Do Developers React to API Evolution? The Pharo Ecosystem Case”. en. In: *Software Maintenance and Evolution (ICSME), 2015 IEEE International Conference On*. IEEE, pp. 251–260. ISBN: 978-1-4673-7532-0. DOI: 10.1109/ICSM.2015.7332471. URL: <http://ieeexplore.ieee.org/document/7332471/> (visited on 05/07/2018).
- Houston, Lara et al. (2016). “Values in Repair”. en. In: *CHI ’16 Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. ACM Press, pp. 1403–1414. ISBN: 978-1-4503-3362-7. DOI: 10.1145/2858036.2858470. URL: <http://dl.acm.org/citation.cfm?doid=2858036.2858470> (visited on 05/01/2018).
- Hu, Yan et al. (Oct. 2018). “User Influence Analysis for Github Developer Social Networks”. en. In: *Expert Systems with Applications* 108, pp. 108–118. ISSN: 09574174. DOI: 10.1016/j.eswa.2018.05.002. URL: <https://linkinghub.elsevier.com/retrieve/pii/S0957417418302793> (visited on 07/16/2019).
- Hughes, Thomas P (1987). “The Evolution of Large Technological Systems”. In: *The Social Construction of Technological Systems: New Directions in the Sociology and History of Technology*. Ed. by Wiebe E Bijker, Thomas P Hughes, and Trevor J Pinch. Cambridge, Mass.: MIT Press, pp. 51–82.
- Humphreys, Lee (Jan. 2005). “Reframing Social Groups, Closure, and Stabilization in the Social Construction of Technology”. en. In: *Social Epistemology* 19.2-3, pp. 231–253. ISSN: 0269-1728, 1464-5297.

- DOI: 10.1080/02691720500145449. URL: <http://www.tandfonline.com/doi/full/10.1080/02691720500145449> (visited on 07/31/2015).
- Hürsch, Walter L. and Cristina Videira Lopes (1995). *Separation of Concerns*. Tech. rep. Technical Report NU- CCS-95-03. Boston, MA: College of Computer Science, Northeastern University.
- Huws, Ursula (2014). *Labor in the Global Digital Economy: The Cybertariat Comes of Age*. New York: Monthly Review Press.
- IndieWeb.org (2015). *Founders*. URL: <https://indieweb.org/founders> (visited on 04/10/2020).
- (2017). *Silo*. URL: <http://indieweb.org/silo> (visited on 04/21/2017).
 - (2018a). *IndieWeb*. URL: <https://indieweb.org/IndieWeb> (visited on 01/28/2019).
 - (2018b). *Plurality*. URL: <https://indieweb.org/plurality> (visited on 04/21/2020).
 - (2018c). *Standards*. URL: <https://spec.indieweb.org/> (visited on 04/20/2020).
 - (2019a). *2011/Guest List - IndieWeb*. URL: https://indieweb.org/2011/Guest_List (visited on 04/10/2020).
 - (2019b). *2018/Organizers - CoC Review and Designated Contacts for Chat*. URL: https://indieweb.org/2018/Organizers%5C#CoC_Review_and_Designated_Contacts_for_Chat (visited on 06/01/2020).
 - (2019c). *Category:Building-Blocks*. URL: <https://indieweb.org/Category:building-blocks> (visited on 04/20/2020).
 - (2019d). *Code-of-Conduct - IndieWeb*. URL: <https://indieweb.org/code-of-conduct> (visited on 03/02/2020).
 - (Sept. 2019e). *Microsub-Spec*. URL: <https://indieweb.org/Microsub-spec> (visited on 03/18/2020).
 - (2019f). *Monoculture*. URL: <http://indieweb.org/monoculture> (visited on 02/08/2017).
 - (2019g). *Organizers - IndieWeb*. URL: <https://indieweb.org/Organizers> (visited on 06/01/2020).
 - (2019h). *Private Webmention*. URL: <https://indieweb.org/Private-Webmention> (visited on 06/16/2020).
 - (2020a). *AutoAuth*. URL: <https://indieweb.org/AutoAuth> (visited on 06/16/2020).
 - (2020b). *Generations*. URL: <https://indieweb.org/generations> (visited on 06/02/2019).
 - (2020c). *Home*. URL: <https://indieweb.org/> (visited on 04/09/2020).
 - (2020d). *Known*. URL: <https://indieweb.org/Known> (visited on 05/24/2020).
 - (2020e). *Longevity - IndieWeb*. URL: <https://indieweb.org/longevity> (visited on 05/01/2020).
 - (2020f). *Micro.Blog - IndieWeb*. URL: <https://indieweb.org/Micro.blog> (visited on 06/03/2020).
 - (2020g). *Microsub*. URL: <https://indieweb.org/Microsub> (visited on 04/20/2020).
 - (2020h). *P3k*. URL: <https://indieweb.org/p3k> (visited on 03/11/2020).
 - (2020i). *Principles*. URL: <https://indieweb.org/principles> (visited on 05/01/2020).
 - (2020j). *Private Posts*. URL: https://indieweb.org/private_posts (visited on 06/16/2020).
 - (2020k). *Webmention*. URL: <https://indieweb.org/Webmention%5C#Services> (visited on 04/21/2020).
 - (2020l). *Why*. URL: <http://indieweb.org/why> (visited on 02/10/2020).
- IndieWebify.Me (2019). *IndieWebify.Me - A Guide to Getting You on the IndieWeb*. URL: <http://indiewebify.me/> (visited on 10/26/2016).
- Ingram, Mathew (Sept. 2014a). *Don't like Facebook Owning and Controlling Your Content? Use Tools That Support the Open Web*. URL: <https://gigaom.com/2014/09/03/dont-like-facebook-owning-and-controlling-your-content-use-tools-that-support-the-open-web/> (visited on 10/16/2016).

- Ingram, Mathew (Sept. 2014b). *IndieWeb Advocates Launch Known so Bloggers Can Be Social and Still Control Their Content*. URL: <https://gigaom.com/2014/09/11/indieweb-advocates-launch-known-so-bloggers-can-be-social-and-still-control-their-content/> (visited on 01/17/2017).
- Jackson, Steven J. (2014). “Rethinking Repair”. In: *Media Technologies: Essays on Communication, Materiality and Society*. Ed. by Tarleton Gillespie, Pablo Boczkowski, and Kirsten Foot. Cambridge: MIT Press.
- JafariNaimi, Nassim, Lisa Nathan, and Ian Hargraves (Oct. 2015). “Values as Hypotheses: Design, Inquiry, and the Service of Values”. en. In: *Design Issues* 31.4, pp. 91–104. ISSN: 0747-9360, 1531-4790. DOI: 10.1162/DESI_a_00354. URL: http://www.mitpressjournals.org/doi/10.1162/DESI_a_00354 (visited on 11/12/2018).
- Jamieson, Jack and Dawn Walker (Sept. 2019). *Remaking the Web: Competing Visions of Decentralization*. Our Networks. Toronto, ON, Canada.
- Joerges, Bernward (June 1999). “Do Politics Have Artefacts?” In: *Social Studies of Science* 29.3, pp. 411–431.
- Johnson, Björn, Edward Lorenz, and Bengt-Åke Lundvall (2002). “Why All This Fuss about Codified and Tacit Knowledge?” In: *Industrial and Corporate Change* 11.2, pp. 245–262. ISSN: 0960-6491. URL: <http://journals.scholarsportal.info/detailsundefined> (visited on 02/06/2017).
- Kahle, Brewster (Aug. 2015). *Locking the Web Open: A Call for a Decentralized Web* —. en-US. URL: <http://brewster.kahle.org/2015/08/11/locking-the-web-open-a-call-for-a-distributed-web-2/> (visited on 01/17/2019).
- Kalliamvakou, Eirini et al. (Oct. 2016). “An In-Depth Study of the Promises and Perils of Mining GitHub”. en. In: *Empirical Software Engineering* 21.5, pp. 2035–2071. ISSN: 1382-3256, 1573-7616. DOI: 10.1007/s10664-015-9393-5. URL: <http://link.springer.com/10.1007/s10664-015-9393-5> (visited on 05/11/2018).
- Kalman, Yoram M. et al. (Oct. 2006). “Pauses and Response Latencies: A Chronemic Analysis of Asynchronous CMC”. en. In: *Journal of Computer-Mediated Communication* 12.1, pp. 1–23. ISSN: 1083-6101, 1083-6101. DOI: 10.1111/j.1083-6101.2006.00312.x. URL: <http://doi.wiley.com/10.1111/j.1083-6101.2006.00312.x> (visited on 11/27/2014).
- Kelty, Christopher M. (2008). *Two Bits: The Cultural Significance of Free Software*. Duke University Press.
- Kennedy, Helen (Dec. 2011). *Net Work: Ethics and Values in Web Design*. English. New York: Palgrave Macmillan. ISBN: 978-0-230-23140-5.
- Khamis, Sahar and Katherine Vaughn (2011). “Cyberactivism in the Egyptian Revolution: How Civic Engagement and Citizen Journalism Tilted the Balance”. In: *Arab Media and Society* 13.3. URL: http://agoraist.com/2012-fall/wp-content/uploads/2013/01/20120313094800_Khamis_Cyberactivism_updated.pdf (visited on 08/04/2013).
- Kickstarter.com (2019). *Indie Microblogging: Owning Your Short-Form Writing*. en. URL: <https://www.kickstarter.com/projects/manton/indie-microblogging-owning-your-short-form-writing> (visited on 04/13/2020).
- Kleiner, Dmitri (2010). *The Telekommunist Manifesto*. Amsterdam: Institute of Network Cultures.
- Kosner, Anthony Wing (2018). *GitHub Is The Next Big Social Network, Powered By What You Do, Not Who You Know*. en. URL: <https://www.forbes.com/sites/anthonykosner/2012/07/15/github->

- is-the-next-big-social-network-powered-by-what-you-do-not-who-you-know/ (visited on 12/20/2018).
- Kuechler, Victor, Claire Gilbertson, and Carlos Jensen (2012). “Gender Differences in Early Free and Open Source Software Joining Process”. en. In: *Open Source Systems: Long-Term Sustainability*. Ed. by Imed Hammouda et al. IFIP Advances in Information and Communication Technology. Berlin, Heidelberg: Springer, pp. 78–93. ISBN: 978-3-642-33442-9. DOI: 10.1007/978-3-642-33442-9_6.
- Kuusinen, Kati et al. (2016). “Flow, Intrinsic Motivation, and Developer Experience in Software Engineering”. In: *Agile Processes, in Software Engineering, and Extreme Programming*. Ed. by Helen Sharp and Tracy Hall. Vol. 251. Cham: Springer International Publishing, pp. 104–117. ISBN: 978-3-319-33514-8 978-3-319-33515-5. DOI: 10.1007/978-3-319-33515-5_9. URL: http://link.springer.com/10.1007/978-3-319-33515-5_9 (visited on 02/27/2020).
- Lambiotte, Renaud et al. (2018). “Beyond Graph Mining: Higher-Order Data Analytics for Temporal Network Data”. en. In: p. 6.
- Lanier, Jaron (Feb. 2011). *You Are Not a Gadget: A Manifesto*. English. Reprint edition. New York: Vintage. ISBN: 978-0-307-38997-8.
- Latour, Bruno (Jan. 2004). “Why Has Critique Run out of Steam? From Matters of Fact to Matters of Concern”. en. In: *Critical Inquiry* 30.2, pp. 225–248. ISSN: 0093-1896, 1539-7858. DOI: 10.1086/421123. URL: <http://www.jstor.org/stable/10.1086/421123> (visited on 11/13/2015).
- (2005). *Reassembling the Social: An Introduction to Actor-Network-Theory*. Clarendon Lectures in Management Studies. New York, New York: Oxford University Press. ISBN: 978-0-19-925604-4.
- (2008). “Where Are the Missing Masses? Sociology of a Few Mundane Artifacts”. In: *Technology and Society, Building Our Sociotechnical Future*. Ed. by Deborah J. Johnson and M Wetmore Jameson. Cambridge, Mass: MIT Press, pp. 151–180.
- Lave, Jean and Etienne Wenger (1991). *Situated Learning: Legitimate Peripheral Participation*. Learning in Doing. Cambridge [England] ; New York: Cambridge University Press. ISBN: 978-0-521-41308-4 978-0-521-42374-8.
- Lima, Antonio, Luca Rossi, and Mirco Musolesi (June 2014). “Coding Together at Scale: GitHub as a Collaborative Social Network”. en. In: *Proceedings of the Eighth International AAAI Conference on Weblogs and Social Media*. Ann Arbor, MI, p. 10.
- Luker, Kristin (2008). *Salsa Dancing into the Social Sciences: Research in an Age of Info-Glut*. Cambridge Massachusetts: Harvard University Press.
- Mabey, Ben (2018). *Bmabey/pyLDavis: Python Library for Interactive Topic Model Visualization. Port of the R LDavis Package*. URL: <https://github.com/bmabey/pyLDavis> (visited on 03/03/2020).
- MacArthur, Cayley, Caroline Wong, and Mark Hancock (Nov. 2019). “Makers and Quilters: Investigating Opportunities for Improving Gender-Imbalanced Maker Groups”. en. In: *Proceedings of the ACM on Human-Computer Interaction* 3.CSCW, pp. 1–24. ISSN: 2573-0142, 2573-0142. DOI: 10.1145/3359131. URL: <https://dl.acm.org/doi/10.1145/3359131> (visited on 05/27/2020).
- Mahan, Kyle (2017). *Kylewm/Woodwind*. URL: <https://github.com/kylewm/woodwind> (visited on 04/23/2020).
- Manders-Huits, Noëmi (June 2011). “What Values in Design? The Challenge of Incorporating Moral Values into Design”. en. In: *Science and Engineering Ethics* 17.2, pp. 271–287. ISSN: 1353-3452, 1471-5546. DOI: 10.1007/s11948-010-9198-2. URL: <http://link.springer.com/article/10.1007/s11948-010-9198-2> (visited on 01/05/2017).

- Marcuse, Herbert (1941). "Some Social Implications of Modern Technology". In: DOI: 10.4324/9780203208311-6.
- Matamoros-Fernández, Ariadna (June 2017). "Platformed Racism: The Mediation and Circulation of an Australian Race-Based Controversy on Twitter, Facebook and YouTube". en. In: *Information, Communication & Society* 20.6, pp. 930–946. ISSN: 1369-118X, 1468-4462. DOI: 10.1080/1369118X.2017.1293130. URL: <https://www.tandfonline.com/doi/full/10.1080/1369118X.2017.1293130> (visited on 09/10/2019).
- Mattingly, Marybeth J. and Liana C. Sayer (2006). "Under Pressure: Gender Differences in the Relationship Between Free Time and Feeling Rushed". en. In: *Journal of Marriage and Family* 68.1, pp. 205–221. ISSN: 1741-3737. DOI: 10.1111/j.1741-3737.2006.00242.x. URL: <http://onlinelibrary.wiley.com/doi/abs/10.1111/j.1741-3737.2006.00242.x> (visited on 06/09/2020).
- McAllan, Ashton (2014). *WhisperFollow*. URL: <https://wordpress.org/plugins/whisperfollow/> (visited on 04/23/2020).
- McCallum, Andrew Kachites (2002). *MALLET: A Machine Learning for Language Toolkit*. URL: <http://mallet.cs.umass.edu/> (visited on 11/12/2019).
- McChesney, Robert W. (2013). *Digital Disconnect: How Capitalism Is Turning the Internet against Democracy*. New Press, The.
- McFadden, Daniel (1978). "Quantitative Methods for Analysing Travel Behaviour of Individuals: Some Recent Developments". In: *Behavioural Travel Modelling*. Ed. by David A. Hensher and Peter R. Stopher. London: Croom Helm London, pp. 279–318. URL: https://eml.berkeley.edu/reprints/mcfadden/quant_methods.pdf.
- McPherson, Miller, Lynn Smith-Lovin, and James M Cook (2001). "Birds of a Feather: Homophily in Social Networks". In: *Annual Review of Sociology* 27.1, pp. 415–444. DOI: 10.1146/annurev.soc.27.1.415. URL: <https://doi.org/10.1146/annurev.soc.27.1.415> (visited on 06/23/2020).
- McVerry, Greg and Jack Jamieson (2018). "Open Pedagogy: Comparative Case Study of #IndieWeb and #ds106". In: *68th Conference of the Literacy Research Association*. Palm Springs, CA.
- Mehrotra, Rishabh et al. (2013). "Improving LDA Topic Models for Microblogs via Tweet Pooling and Automatic Labeling". en. In: ACM Press, p. 889. ISBN: 978-1-4503-2034-4. DOI: 10.1145/2484028.2484166. URL: <http://dl.acm.org/citation.cfm?doid=2484028.2484166> (visited on 05/28/2018).
- Microformats.org (2019). *H-Event · Microformats Wiki*. URL: <http://microformats.org/wiki/h-event> (visited on 04/28/2020).
- Miller, Daniel and Don Slater (2003). *The Internet: An Ethnographic Approach*. Oxford ; New York: Berg.
- Miller, Jessica K. et al. (2007). "Value Tensions in Design: The Value Sensitive Design, Development, and Appropriation of a Corporation's Groupware System". In: *Proceedings of the 2007 International ACM Conference on Supporting Group Work*. GROUP '07. New York, NY, USA: ACM, pp. 281–290. ISBN: 978-1-59593-845-9. DOI: 10.1145/1316624.1316668. URL: <http://doi.acm.org/10.1145/1316624.1316668> (visited on 02/07/2017).
- Moglen, Eben (2003). *The dotCommunist Manifesto*. URL: http://emoglen.law.columbia.edu/my_pubs/dcm.html (visited on 04/13/2020).

- Montfort, Nick and Ian Bogost (2009). *Racing the Beam: The Atari Video Computer System*. Platform Studies. Cambridge: MIT Press. URL: <https://mitpress.mit.edu/books/racing-beam> (visited on 01/17/2017).
- Moon, Brenda (Oct. 2016). “A Topic Analysis Approach to Revealing Discussions on the Australian Twittersphere”. en. In: *Selected Papers of AoIR 2016: The 17th Annual Conference of Internet Researchers*. Berlin, Germany, p. 4.
- Mosco, Vincent (2009). *The Political Economy of Communication*. 2nd ed. Los Angeles: SAGE. ISBN: 978-1-4129-4700-8 978-1-4129-4701-5.
- Mutton, Paul (2004). “Inferring and Visualizing Social Networks on Internet Relay Chat”. In: *Information Visualisation, 2004. IV 2004. Proceedings. Eighth International Conference On*. IEEE, pp. 35–43. URL: <http://ieeexplore.ieee.org/abstract/document/1320122/> (visited on 07/18/2017).
- Nafus, Dawn (June 2012). “‘Patches Don’t Have Gender’: What Is Not Open in Open Source Software”. en. In: *New Media & Society* 14.4, pp. 669–683. ISSN: 1461-4448. DOI: 10.1177/1461444811422887. URL: <https://doi.org/10.1177/1461444811422887> (visited on 03/02/2020).
- Narayanan, Arvind et al. (Feb. 2012). “A Critical Look at Decentralized Personal Data Architectures”. In: *arXiv:1202.4503 [cs]*. arXiv: 1202.4503 [cs]. URL: <http://arxiv.org/abs/1202.4503> (visited on 02/18/2020).
- Neff, Gina (2012). *Venture Labor: Work and the Burden of Risk in Innovative Industries*. Acting with Technology. Cambridge, Mass: MIT Press. ISBN: 978-0-262-01748-0.
- NetworkX (2019). *NetworkX*. URL: <https://networkx.github.io/> (visited on 03/04/2020).
- Nguyen, Dat Quoc et al. (n.d.). “Improving Topic Models with Latent Feature Word Representations”. en. In: (), p. 16.
- Norman, Donald A. (1999). “Affordance, Conventions, and Design”. In: *interactions* 6.3, pp. 38–43. URL: <http://dl.acm.org/citation.cfm?id=301168> (visited on 11/18/2014).
- O’Reilly, Tim (Sept. 2005). *What Is Web 2.0*. URL: <http://www.oreilly.com/pub/a/web2/archive/what-is-web-20.html?page=1> (visited on 06/09/2015).
- (2007). “What Is Web 2.0: Design Patterns and Business Models for the next Generation of Software”. In: *Communications & strategies* 1, p. 17. URL: http://papers.ssrn.com/sol3/Papers.cfm?abstract_id=1008839 (visited on 08/24/2015).
- Orr, Julian E. (1996). *Talking about Machines: An Ethnography of a Modern Job*. Collection on Technology and Work. Ithaca, N.Y: ILR Press. ISBN: 978-0-8014-3297-2 978-0-8014-8390-5.
- Parecki, Aaron (Jan. 2017). *Webmention*. en. URL: <https://www.w3.org/TR/webmention/> (visited on 12/15/2017).
- (June 2018). *Social Readers - IndieWeb Summit 2018*. Portland, OR. URL: <https://www.youtube.com/watch?v=inhgyhx0vt4%5C&list=PLk3TtIJ31hqodLN698UsRXsOX-WqLneJ6%5C&index=4> (visited on 04/28/2020).
- (Feb. 2019). *Getting Started with Readers*. Austin, Texas, USA. URL: <https://www.youtube.com/watch?v=H5oZLqevQ6g%5C&t=53m03s> (visited on 04/30/2020).
- Pariser, Eli (2012). *The Filter Bubble: How the New Personalized Web Is Changing What We Read and How We Think*. English. ISBN: 978-1-322-77515-9. URL: <http://www.myilibrary.com?id=708797> (visited on 10/20/2018).

- Phan, X. et al. (July 2011). “A Hidden Topic-Based Framework toward Building Applications with Short Web Documents”. In: *IEEE Transactions on Knowledge and Data Engineering* 23.7, pp. 961–976. ISSN: 1041-4347. DOI: 10.1109/TKDE.2010.27.
- Pinch, Trevor J. and Wiebe E Bijker (1987). “The Social Construction of Facts and Artifacts: Or How the Sociology of Science and the Sociology of Technology Might Benefit Each Other”. en. In: *The Social Constructions of Technological Systems: New Directions in the Sociology and History of Technology*. Ed. by Thomas P. Hughes, Wiebe E. Bijker, and Trevor J. Pinch. Cambridge, Mass.: MIT Press, pp. 17–50. ISBN: 978-0-262-26044-2.
- Plantin, Jean-Christophe (2017). “Mapping Platforms as Infrastructures: Participatory Cartography, Enclosed Knowledge”. en. In: *International Journal of Communication*, p. 30.
- Plantin, Jean-Christophe, Carl Lagoze, et al. (2016). “Infrastructure Studies Meet Platform Studies in the Age of Google and Facebook”. In: *new media & society*, pp. 1–18. URL: <http://nms.sagepub.com/content/early/2016/08/02/1461444816661553.abstract> (visited on 09/27/2016).
- Plantin, Jean-Christophe and A. Powell (2016). “Open Maps, Closed Knowledge: What the Platformization of Maps Means for Citizenship and Society”. In: *IPP2016: The Platform Society*. Oxford.
- Prabhakaran, Selva (Mar. 2018). *Topic Modeling in Python with Gensim*. en-US. URL: <https://www.machinelearningplus.com/nlp/topic-modeling-gensim-python/> (visited on 03/03/2020).
- PyGitHub (Mar. 2020). *PyGitHub*. PyGithub. URL: <https://github.com/PyGithub/PyGithub> (visited on 03/02/2020).
- Raggett, Dave (Jan. 1997). *HTML 3.2 Reference Specification*. URL: <https://www.w3.org/TR/REC-html32> (visited on 01/11/2017).
- Raggett, Dave et al. (1998). *Raggett on HTML 4*. English. 2 edition. Harlow, England: Addison Wesley Longman. ISBN: 978-0-201-17805-2. URL: <http://www.w3.org/People/Raggett/book4/ch01.html>.
- Randall, Dave, Richard Harper, and Mark Rouncefield (2007). *Fieldwork for Design*. en. Springer.
- Ratto, Matt (2009). “Epistemic Commitments, Virtual Reality, and Archaeological Representation”. In: *Proceedings of the International Union for Prehistoric and Protohistoric Sciences XV*. Lisbon, Portugal.
- (July 2011). “Critical Making: Conceptual and Material Studies in Technology and Social Life”. In: *The Information Society* 27.4, pp. 252–260. ISSN: 0197-2243. DOI: 10.1080/01972243.2011.583819. URL: <http://dx.doi.org/10.1080/01972243.2011.583819> (visited on 10/05/2016).
- (2014). “Textual Doppelgangers: Critical Issues in the Study of Technology”. In: *DIY Citizenship: Critical Making and Social Media*. Ed. by Matt Ratto and Megan Boler. Cambridge Massachusetts: The MIT Press, pp. 227–236. ISBN: 978-0-262-02681-9 978-0-262-52552-7.
- (2016). “Making at the End of Nature”. In: *interactions* 23.5, pp. 26–35. URL: <http://dl.acm.org/citation.cfm?id=2985851> (visited on 02/21/2017).
- (forthcoming). *Critical Making*. Cambridge, MA: MIT Press.
- Reagle, Joseph (2013). ““Free as in Sexist?” Free Culture and the Gender Gap”. en. In: *First Monday*. ISSN: 1396-0466. DOI: 10.5210/fm.v18i1.4291. URL: <https://firstmonday.org/ojs/index.php/fm/article/view/4291> (visited on 03/02/2020).
- Reece, Manton (July 2020). *[Personal Correspondance]*.
- Rehurek, Radim and Petr Sojka (2010). “Software Framework for Topic Modelling with Large Corpora”. In: *In Proceedings of the Lrec 2010 Workshop on New Challenges for Nlp Frameworks*, pp. 45–50.

- Richard, Gabriela T. and Kishonna L. Gray (Apr. 2018). “Gendered Play, Racialized Reality: Black Cyberfeminism, Inclusive Communities of Practice, and the Intersections of Learning, Socialization, and Resilience in Online Gaming”. en. In: *Frontiers: A Journal of Women Studies* 39.1, pp. 112–148. ISSN: 1536-0334. URL: <http://muse.jhu.edu/article/690812> (visited on 06/07/2020).
- Roberts, Benjamin and Kevin Marks, eds. (Nov. 2018). *JF2 Post Serialization Format*. URL: <https://www.w3.org/TR/jf2/%5C#jf2feed> (visited on 04/29/2020).
- Röder, Michael, Andreas Both, and Alexander Hinneburg (2015). “Exploring the Space of Topic Coherence Measures”. en. In: *Proceedings of the Eighth ACM International Conference on Web Search and Data Mining - WSDM '15*. Shanghai, China: ACM Press, pp. 399–408. ISBN: 978-1-4503-3317-7. DOI: 10.1145/2684822.2685324. URL: <http://dl.acm.org/citation.cfm?doid=2684822.2685324> (visited on 05/22/2020).
- Roedl, David, Shaowen Bardzell, and Jeffrey Bardzell (June 2015). “Sustainable Making? Balancing Optimism and Criticism in HCI Discourse”. In: *ACM Transactions on Computer-Human Interaction* 22.3, 15:1–15:27. ISSN: 1073-0516. DOI: 10.1145/2699742. URL: <http://doi.org/10.1145/2699742> (visited on 03/11/2020).
- Roosendaal, Arnold (2011). “Facebook Tracks and Traces Everyone: Like This!” In: *Tilburg Law School Legal Studies Research Paper Series* 03. URL: http://papers.ssrn.com/sol3/papers.cfm?abstract_id=1717563 (visited on 10/17/2015).
- Saint-Louis, Hervé (2018). “User Perceptions of Security Risks in Multiple Authentications”. en. PhD thesis. Toronto, ON, Canada: University of Toronto.
- Saldaña, Johnny (2013). *The Coding Manual for Qualitative Researchers*. en. 2nd ed. Los Angeles: SAGE. ISBN: 978-1-4462-4736-5 978-1-4462-4737-2.
- Sandvig, Christian (Mar. 2013). “The Internet as Infrastructure”. In: *The Oxford Handbook of Internet Studies*. Ed. by William H. Dutton. Oxford: Oxford University Press.
- Sayer, L. C. (Sept. 2005). “Gender, Time and Inequality: Trends in Women’s and Men’s Paid Work, Unpaid Work and Free Time”. en. In: *Social Forces* 84.1, pp. 285–303. ISSN: 0037-7732, 1534-7605. DOI: 10.1353/sof.2005.0126. URL: <https://academic.oup.com/sf/article-lookup/doi/10.1353/sof.2005.0126> (visited on 06/09/2020).
- Schäfer, Mirko Tobias (2011). *Bastard Culture!: How User Participation Transforms Cultural Production*. en. Amsterdam: Amsterdam University Press. ISBN: 978-90-8964-256-1.
- Schneider, Nathan (Apr. 2019). “Decentralization: An Incomplete Ambition”. In: *Journal of Cultural Economy* 0.0, pp. 1–21. ISSN: 1753-0350. DOI: 10.1080/17530350.2019.1589553. URL: <https://doi.org/10.1080/17530350.2019.1589553> (visited on 08/12/2019).
- Scholz, Trebor and Nathan Schneider, eds. (2016). *Ours to Hack and to Own: The Rise of Platform Cooperativism, a New Vision for the Future of Work and a Fairer Internet*. English. New York: OR BOOKS. ISBN: 978-1-68219-062-3.
- Schwartz, Shalom (Dec. 2012). “An Overview of the Schwartz Theory of Basic Values”. In: *Online Readings in Psychology and Culture* 2.1. ISSN: 2307-0919. DOI: 10.9707/2307-0919.1116. URL: <https://scholarworks.gvsu.edu/orpc/vol2/iss1/11>.
- Schwartz, Shalom H (1992). “Universals in the Content and Structure of Values: Theory and Empirical Tests in 20 Countries”. In: *Advances in experimental social psychology* 25, pp. 1–65.
- scikit-learn (2020). *Scikit-Learn: Machine Learning in Python*. URL: <https://scikit-learn.org/stable/> (visited on 03/04/2020).

- Senft, Theresa M. and Nancy K. Baym (Sept. 2015). “What Does the Selfie Say? Investigating a Global Phenomenon”. English. In: *International Journal of Communication* 9.Feature, pp. 1588–1606. ISSN: 1932-8036. URL: <https://researchers.mq.edu.au/en/publications/what-does-the-selfie-say-investigating-a-global-phenomenon> (visited on 07/09/2020).
- Shilton, Katie, Jes A. Koepfler, and Kenneth R. Fleischmann (2013). “Charting Sociotechnical Dimensions of Values for Design Research”. In: *The Information Society* 29.5, pp. 259–271. URL: <http://www.tandfonline.com/doi/abs/10.1080/01972243.2013.825357> (visited on 06/09/2015).
- Shirky, Clay (Feb. 2008). *Here Comes Everybody: The Power of Organizing without Organizations*. en. New York: Penguin. ISBN: 978-1-4406-3224-2.
- (June 2010). *Cognitive Surplus: How Technology Makes Consumers into Collaborators*. en. New York: Penguin. ISBN: 978-1-101-43472-7.
- Shore, Lynn M. et al. (June 2009). “Diversity in Organizations: Where Are We Now and Where Are We Going?” en. In: *Human Resource Management Review*. Emerging Trends in Human Resource Management Theory and Research 19.2, pp. 117–133. ISSN: 1053-4822. DOI: 10.1016/j.hrmr.2008.10.004. URL: <http://www.sciencedirect.com/science/article/pii/S1053482208000855> (visited on 06/23/2020).
- Sievert, Carson and Kenneth Shirley (2014). “LDAvis: A Method for Visualizing and Interpreting Topics”. en. In: *Proceedings of the Workshop on Interactive Language Learning, Visualization, and Interfaces*. Baltimore, Maryland, USA: Association for Computational Linguistics, pp. 63–70. DOI: 10.3115/v1/W14-3110. URL: <http://aclweb.org/anthology/W14-3110> (visited on 08/20/2019).
- Singh, Rianka (Nov. 2018). “Platform Feminism: Protest and the Politics of Spatial Organization”. English. In: *Ada New Media* 14. URL: <https://adanewmedia.org/2018/11/issue14-singh/> (visited on 02/06/2020).
- Smith, Aaron and Monica Anderson (Mar. 2018). *Social Media Use 2018*. en-US. Tech. rep. Pew Research Center. URL: <http://www.pewinternet.org/2018/03/01/social-media-use-in-2018/> (visited on 01/27/2019).
- Söderberg, Johan (2008). *Hacking Capitalism: The Free and Open Source Software Movement*. en. Routledge Research in Information Technology and Society 9. New York: Routledge. ISBN: 978-0-415-95543-0.
- Song, Felicia Wu (Mar. 2010). “Theorizing Web 2.0: A Cultural Perspective”. en. In: *Information, Communication & Society* 13.2, pp. 249–275. ISSN: 1369-118X, 1468-4462. DOI: 10.1080/13691180902914610. URL: <http://www.tandfonline.com/doi/abs/10.1080/13691180902914610> (visited on 08/11/2015).
- spaCy (2020). *spaCy · Industrial-Strength Natural Language Processing in Python*. en. URL: <https://spacy.io/> (visited on 03/03/2020).
- Spencer, Steven J., Christine Logel, and Paul G. Davies (2016). “Stereotype Threat”. In: *Annual Review of Psychology* 67.1, pp. 415–437. DOI: 10.1146/annurev-psych-073115-103235. URL: <https://doi.org/10.1146/annurev-psych-073115-103235> (visited on 06/07/2020).
- Srnicek, Nick (2017). *Platform Capitalism*. Cambridge, UK: Polity Press.
- Star, Susan Leigh (1999). “The Ethnography of Infrastructure”. In: *American Behavioral Scientist* 43.3, pp. 377–391.
- (Sept. 2010). “This Is Not a Boundary Object: Reflections on the Origin of a Concept”. en. In: *Science, Technology & Human Values* 35.5, pp. 601–617. ISSN: 0162-2439, 1552-8251. DOI: 10.1177/

0162243910377624. URL: <http://sth.sagepub.com/cgi/doi/10.1177/0162243910377624> (visited on 02/22/2016).
- Statistics Canada (Aug. 2018). *Labour Force Survey, June 2018 [Canada]*. Tech. rep. lfs-71M0001-E-2018-june. Labour Statistics Division, Statistics Canada.
- Sterling, Bruce (Dec. 2013). “Indieweb Principles”. en-US. In: *Wired*. ISSN: 1059-1028. URL: <https://www.wired.com/2013/12/indieweb-principles/> (visited on 03/10/2020).
- Strzalkowski, Tomek et al. (2019). “GitHub as a Social Network”. en. In: *Advances in Artificial Intelligence, Software and Systems Engineering*. Ed. by Tareq Z. Ahram. Vol. 787. Cham: Springer International Publishing, pp. 379–390. ISBN: 978-3-319-94228-5 978-3-319-94229-2. DOI: 10.1007/978-3-319-94229-2_37. URL: http://link.springer.com/10.1007/978-3-319-94229-2_37 (visited on 07/16/2019).
- Suchman, Lucy (1997). “Do Categories Have Politics? The Language/Action Perspective Reconsidered”. In: *Human Values and the Design of Computer Technology*. Ed. by Batya Friedman. CSLI Lecture Notes no. 72. New York: Cambridge University Press, pp. 91–106. ISBN: 978-1-57586-081-7 978-1-57586-080-0.
- (2002). “Located Accountabilities in Technology Production”. en. In: 14, p. 16.
- (2007). *Human-Machine Reconfigurations: Plans and Situated Actions*. English. Cambridge; New York: Cambridge University Press. ISBN: 978-0-521-85891-5 978-0-521-67588-8 978-0-511-25592-2 978-0-511-25649-3.
- Tang, John C. (1997). “Eliminating a Hardware Switch: Weighing Economics and Values in a Design Decision”. In: *Human Values and the Design of Computer Technology*. Ed. by Batya Friedman. CSLI Lecture Notes no. 72. New York: Cambridge University Press, pp. 259–270. ISBN: 978-1-57586-081-7 978-1-57586-080-0.
- Taylor, Nick, Ursula Hurley, and Philip Connolly (May 2016). “Making Community: The Wider Role of Makerspaces in Public Life”. en. In: *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. San Jose California USA: ACM, pp. 1415–1425. ISBN: 978-1-4503-3362-7. DOI: 10.1145/2858036.2858073. URL: <https://dl.acm.org/doi/10.1145/2858036.2858073> (visited on 05/27/2020).
- Terranova, Tiziana (2000). “Free Labor: Producing Culture for the Digital Economy”. In: *Social text* 18.2, pp. 33–58. URL: <http://muse.jhu.edu/journals/soc/summary/v018/18.2terranova.html> (visited on 11/04/2015).
- Turkle, Sherry (2011). *Alone Together: Why We Expect More from Technology and Less from Each Other*. English. New York: Basic Books. ISBN: 978-0-465-02234-2 0-465-02234-0.
- Tuulos, Ville H. and Henry Tirri (2004). “Combining Topic Models and Social Networks for Chat Data Mining”. In: *Proceedings of the 2004 IEEE/WIC/ACM International Conference on Web Intelligence*. IEEE Computer Society, pp. 206–213.
- van Wendel de Joode, Ruben (June 2004). “Managing Conflicts in Open Source Communities”. en. In: *Electronic Markets* 14.2, pp. 104–113. ISSN: 1019-6781, 1422-8890. DOI: 10.1080/10196780410001675059. URL: <http://www.informaworld.com/openurl?genre=article%5C&doi=10.1080/10196780410001675059%5C&magic=crossref%7C%7CD404A21C5BB053405B1A640AFFD44AE3> (visited on 05/28/2020).
- Vasilescu, Bogdan, Andrea Capiluppi, and Alexander Serebrenik (Dec. 2012). “Gender, Representation and Online Participation: A Quantitative Study of StackOverflow”. In: *2012 International Conference on Social Informatics*, pp. 332–338. DOI: 10.1109/SocialInformatics.2012.81.

- von Krogh, Georg et al. (2012). “Carrots and Rainbows: Motivation and Social Practice in Open Source Software Development”. In: *MIS Quarterly* 36.2, pp. 649–676. ISSN: 0276-7783. DOI: 10.2307/41703471. URL: <https://www.jstor.org/stable/41703471> (visited on 03/10/2020).
- W3C (July 2014). *W3C Launches Push for Social Web Application Interoperability — W3C News*. en-US. URL: <https://www.w3.org/blog/news/archives/3958> (visited on 04/20/2020).
- Wenger, Etienne (1998). *Communities of Practice: Learning, Meaning, and Identity*. Cambridge, UK: Cambridge University Press. URL: <https://www-cambridge-org.myaccess.library.utoronto.ca/core/books/communities-of-practice/724C22A03B12D11DFC345EEF0AD3F22A> (visited on 06/07/2020).
- (2011). “Communities of Practice: A Brief Introduction”. In: URL: <https://scholarsbank.uoregon.edu/xmlui/handle/1794/11736> (visited on 07/31/2015).
- Wermüller, Ben (Apr. 2018). *I’m Done with Syndication. Let’s Help People Be Themselves on the Web*. en. URL: <https://werd.io/2018/im-done-with-syndication-lets-help-people-be-themselves-on> (visited on 05/07/2018).
- Whelan, Tara (2018). “We Are Not All Makers: The Paradox of Plurality In The Maker Movement”. en. In: *Proceedings of the 19th International ACM SIGACCESS Conference on Computers and Accessibility - DIS ’18*. Hong Kong, China: ACM Press, pp. 75–80. ISBN: 978-1-4503-5631-2. DOI: 10.1145/3197391.3205415. URL: <http://dl.acm.org/citation.cfm?doid=3197391.3205415> (visited on 05/27/2020).
- Whittle, Jon (May 2019). “Is Your Software Valueless?” en. In: *IEEE Software* 36.3, pp. 112–115. ISSN: 0740-7459, 1937-4194. DOI: 10.1109/MS.2019.2897397. URL: <https://ieeexplore.ieee.org/document/8693084/> (visited on 08/22/2019).
- Wilkinson, David and Mike Thelwall (Nov. 2011). “Researching Personal Information on the Public Web: Methods and Ethics”. en. In: *Social Science Computer Review* 29.4, pp. 387–401. ISSN: 0894-4393, 1552-8286. DOI: 10.1177/0894439310378979. URL: <http://journals.sagepub.com/doi/10.1177/0894439310378979> (visited on 02/04/2020).
- Winner, Langdon (1980). “Do Artifacts Have Politics?” In: *Daedalus*. URL: <http://www.jstor.org/stable/10.2307/20024652>.
- Wirdum, Aaron van (July 2016). *Urbite: The Bold Pitch to Re-Decentralize the Internet, on Top of the Internet*. URL: <https://bitcoinmagazine.com/articles/urbite-the-bold-pitch-to-re-decentralize-the-internet-on-top-of-the-internet-1468257340> (visited on 12/28/2016).
- Wu, Eva Yiwei, Emily Pedersen, and Niloufar Salehi (Nov. 2019). *Agent, Gatekeeper, Drug Dealer: How Content Creators Craft Algorithmic Personas*. URL: <http://doi.org/10.1145/3359321> (visited on 03/27/2020).
- Xavier, Laerte et al. (Feb. 2017). “Historical and Impact Analysis of API Breaking Changes: A Large-Scale Study”. en. In: *Software Analysis, Evolution and Reengineering (SANER), 2017 IEEE 24th International Conference On*. Klagenfurt, Austria: IEEE, pp. 138–147. ISBN: 978-1-5090-5501-2. DOI: 10.1109/SANER.2017.7884616. URL: <http://ieeexplore.ieee.org/document/7884616/> (visited on 05/07/2018).
- Zuboff, Shoshana (2019). *The Age of Surveillance Capitalism: The Fight for a Human Future at the New Frontier of Power*. New York, NY: Hachette Book Group.

Zuckerberg, Mark (Apr. 2018). *Hearing Before the United States House of Representatives Committee on Energy and Commerce*. Washington, D.C., United States. URL: <https://docs.house.gov/meetings/IF/IF02/20180508/108260/HHRG-115-IF02-Wstate-BarrettG-20180508.pdf>.