Stay Connected

A genealogical investigation of the history of social network technologies

Stijn Peeters ◆ MA Thesis

Abstract

Social Network Sites have experienced a rapid increase in popularity since their inception in the latter half of the 1990s. These sites however have also drawn heavy criticism; for example, they have been described as a threat to user's privacy and as not adequately addressing their users' needs. As a response to this alternative social networks have been started - however these enjoy relatively little success and have in turn been criticized for not addressing concerns effectively.

Social Network Sites as they are known now have only existed since 1997 - as described by boyd and Ellison - and were preceded by a variety of other social technologies that often adopted a different paradigm with regards to software development and methods of user involvement. In this thesis, these older technologies (specifically IRC, USENET and Finger) will be described genealogically, to investigate whether their features and paradigms would be of use in addressing the issues with contemporary Social Network Sites. In this analysis the ecosystem – technological, social and political - in which these technologies were developed is key, and an important factor in determining how features found in older technologies would fare in the current ecosystem.

Metadata

Keywords: social network sites, irc, finger, usenet, alternative SNSs, genealogy

Supervisor: Sanne Koevoets

Master Thesis for the New Media Digital Culture MA Programme,
Utrecht University — Utrecht, 15 September 2013

Stijn Peeters, 3503941

Cover illustration: Screenshot of an IRC chat in the channel #whatwg, on the Freenode network.

Table of contents

This is for everyone #london2012 #oneweb

#openingceremony @webfoundation @w3c

@tim_bernerslee, 27 July 2012

Foreword and acknowledgements

"Stay connected!" — I've said it so often, mocking friends when they couldn't resist the temptation to check their Facebook or Twitter updates while going out for a drink. Only when I'd been working on this thesis for a while it began to dawn on me that there might be more to this phrase than using it to casually tease my friends. After all, if there's one thing I've learnt while writing this it is that the internet is a way for people to talk to each other first and foremost. If there's a way to send bits from one computer to the other, people will use it to write an application — no matter how crude — to send messages to one another.

Hopefully this thesis can shed a bit more light on how such applications have evolved throughout the past decades. There's more platforms out there than Facebook, Twitter and their derivatives —some long forgotten, some still going strong in obscure corners of the net. And the early internet was a very different place than the global, ubiquitous, commercial platform we have to deal with nowadays — a platform that has its fair share of issues. Could the solution to these problems lie in the past, when these problems did not exist yet, at least not in the same form? Let's find out.

A thousand thanks to those who have helped me in one way or another in writing this thesis; in particular those who were so generous with their time as to proofread it. I am similarly grateful to the people at the Institute of Network Cultures in Amsterdam, who planted the seed that grew into this thesis during my internship there. I should also thank my supervisor, Sanne Koevoets, for her valuable feedback that helped me put the pieces together. And finally, thanks to Justin, for giving me an excuse to keep mentioning the title of this thesis over and over again.

Stijn Peeters — Utrecht, September 2013

Introduction

Hardly a month goes by without some sort of scandal involving social network sites. The better part of 2013, Facebook and other major social websites have faced criticisms over alleged backdoors in their platforms that allow government agencies easy access to user's data[1] — but before that there were other privacy scandals, users scolding Facebook for not listening to their pleas for new features[2], Twitter facing criticisms over making it difficult for custom clients to access their network[3] — quiet moments are few and far between.

Social network sites are in a peculiar position. On the one hand, a significant portion of the world's population can be found on large platforms like Facebook or Twitter; on the other hand, it would seem that these same social network sites are the target of a never-ending torrent of criticism about how risky it is to trust these sites with personal data, how they never listen to their users and are guided by commercial interests, and how they are redefining our very conception of friendship — and not for the better.

Some protesters have taken matters in their own hands and started their own social network sites; often modelled after Facebook or Twitter's template, but with a few tweaks in those areas that seek to address the concerns the dominance of these social network behemoths have raised, following their

---

1 This criticism was a result of NSA whistleblower Edward Snowden's revelations; see for example http://tweakers.net/nieuws/89559/nsa-en-fbi-hebben-directe-toegang-tot-servers-grote-techbedrijven.html

2 See for example http://scobleizer.com/2009/03/21/why-facebook-has-never-listened-and-why-it-definitely-wont-start-now/

3 See for example http://semiaccurate.com/2011/03/12/twitter-doesnt-like-custom-twitter-clients/

rapid rise to ubiquity[4]. Yet these alternatives face their own problems, and are often only marginally different from mainstream; differences are often technical in nature and do not impact the actual user experience. More fundamental problems like the lack of user influence on the platform's features, or concerns about privacy are often not adequately addressed[5]. So how can alternative networks be made to address such issues with contemporary social network sites more effectively?

One way to improve the present is to learn from the past; so let us look towards the past of social network sites. While things have certainly been developing fast on the social front recently, internet has been a platform for social interaction from the very beginning. From the simplistic TALK command in Unix from the early 1970s to the elaborate etiquette of IRC chat channels in the 1990s, the net has always been a place of lively discussion and community building. And while the technologies that made this possible have been covered in academic writing often enough, there seems to be something of a gap concerning the transition from these often ancient technologies to the Facebooks and Twitters of the modern-day world.

Yet these contemporary powerhouses of online social interaction didn't spring up overnight, and didn't invent the idea of connecting to people via the internet by themselves. They too stood on the shoulders of giants, giants who made the internet a lively meeting place before Mark Zuckerberg was even born. It is therefore worthwhile to investigate how these forerunners approached the problem of how to create a platform for people to interact online; they certainly did a lot of things differently than the behemoths of

---

4 For example, Diaspora (http://joindiaspora.org), Lorea (http://n-1.cc) or Status.net (http://status.net/)

5 See for example Hui and Halpin (2012), Peeters (2013) and Narayanan et al. (2012) for a more exhaustive overview of some prominent issues with these alternative social networks.

today do them, if only because the technology those platforms use wasn't invented yet. At the same time, this is perhaps precisely why they are somewhat forgotten nowadays; we live in a new era of internet usage, with new technological possibilities and new demands from users. So before one would blindly jump to old technologies to see how they would work in today's ecosystem, it is necessary to also consider the *old* ecosystem in which these *old* technologies were developed.

In this thesis I will therefore investigate what distinguishes older technologies for social network activity over the internet from contemporary social network sites like Facebook and Twitter, and what the technological, political or social reasons for these differences may be. Can smaller, alternative social networks, that try to avoid the issues these mainstream platforms have, benefit from adopting some of these older paradigms and features? I will investigate this question by first discussing the history of technologies of computer-mediated communication, or CMC, which is an umbrella term that includes most online social network technologies. After this cursory exploration I return to contemporary social network sites, which have been defined and described in chief by boyd and Ellison; I will investigate how their analysis, which mostly deals with technologies developed after 1997, relates to older CMC technologies, and what older technologies are most akin to newer ones and would therefore benefit from a closer investigation.

CMC technologies, or any technology for that matter, never exists in a vacuum. This closer investigation therefore will not simply put features side-by-side and compare how they differ in user experience or technological background, but rather approach the analysis from a genealogical — in the Foucauldian sense — angle to try to see why these technologies came to be as they were, and how the ecosystem they were developed in shaped them. This should give a clear picture of how these technologies would be of use for addressing

today's problems with social network sites; next, I will therefore discuss three commonly cited issues and see how the findings from the earlier analysis might contribute to addressing these issues. Finally I discuss, based on these findings, to what extent older social network technologies can in general be said to be a worthwhile avenue of investigation for dealing with today's issues with social network sites.

1 Computer-mediated communication and social network technologies

1.1 Computer-mediated communication: a short overview

When analysing social network sites as a form of computer-mediated communication, or CMC, it is important to take a closer look at what CMC exactly is, then, and in what way social network sites relate to it. CMC is a broad concept, and has been around since before the internet existed, so to reconcile it with the more recent concept of social media might take some effort.

CMC is a category of communication technology that typically includes about any technology that uses some form of a computer in its systems. This obviously includes methods of communication such as e-mail, forums and chat (Herring 2005, 112) but also systems like voice mail (Steinfield 1986, 168) and game-like platforms such as MUDs (Jones 1998, xvi). As Steven Jones mentions in his preface to the first *CyberSociety* anthology, CMC has been around for as long as computers have been networked; as soon as university computers in the USA were connected through the internet's predecessor ARPANET, its most important use was "[sharing] information by way of electronic mail" (1995, 3). Such electronic mail, which was still largely analogous to offline mail (one-to-one communication by means of individual messages) was soon

expanded with derivations such as mailing lists and BBSs (Bulletin Board Systems), which allowed for a more communal way of communication (ibid.) in which a message was received by multiple people at once. As the early internet had only a limited amount of users, this often meant that one e-mail could quite literally reach everyone that was connected to it (Jones 1998 xiv; Hafner and Lyon 1998, 189).

As internet usage grew, more and more mailing lists and bulletin boards were set up, and increasingly often these were centred around certain topics rather than being a place for more general-purpose discussion. This culminated in USENET, a "newsgroup" platform (Naughton 1999, 178; fig. 1). Newsgroups were bulletin board-like collections of messages about a specific subject, offered in a "threaded" format where related messages were automatically linked together. The main difference between USENET and earlier forms of electronic group communication were that on USENET, each newsgroup was part of the larger hierarchy; their place in this hierarchy could easily be inferred by the group's name. soc.religion.christian.bible-study for example designated a group about study of the holy scripture, for Christians, within the larger group of religious newsgroups, in the "soc" or "social" hierarchy. alt.binaries.pictures.erotica meanwhile contained erotic pictures, offered in a binary file format, within the "alternative" hierarchy, which had an "anything goes" policy (181).

This hierarchical organization of newsgroups marks a shift in CMC paradigms. Whereas newsgroups were decentralized, without a central directory in which they were collected, on USENET one could easily jump from a discussion on whether Lee Harvey Oswald really killed John F. Kennedy in alt.conspiracies.jfk to musings about medieval heraldry in rec.heraldry. To see which other groups were available, one could simply instruct their news reader program to show a list of groups within a specific part of the hierarchy. The fact that each

newsgroup had a topic designation, coupled with the way that each also had a clear place in the overall structure, gave all discussions and messages a distinct context. Furthermore, as users on a USENET server typically stuck with one account for all their activity on that server, a user's contributions could be traced even if no further personal information was offered (Kolko and Reid 1998, 227). This gave each user a certain identity, and people might learn to look out for a specific user's postings in some newsgroups based on their prior activity. In other words, each message had a clear context in several ways; by being posted within a specific group with a clear place in the overall hierarchy, by being posted by a specific individual, and by being part of a "comment thread" in which messages that were replies to each other were grouped together.

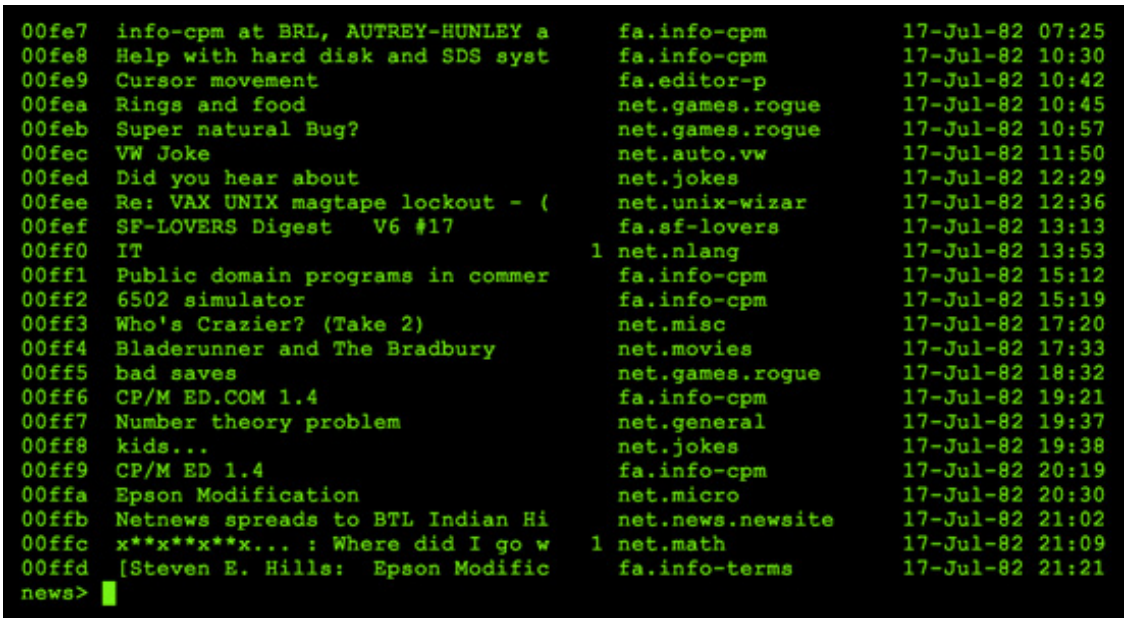

Fig. 1 — A list of recently updated threads on USENET, as seen in the *Telehack* client

USENET was one of the more popular means of CMC well into the 1990s, together with the perpetually popular e-mail. It was however slowly replaced by the World Wide Web, a newer internet platform that was introduced in 1989 at the European CERN research institute (Jones 1998 xv; Naughton 1999, 149). The WWW could potentially offer the same possibilities as older

technologies like USENET, but was more flexible with regards to what features were offered, and better suited to displaying images, custom typography and layout. Additionally, the WWW was one of the first widely adopted thorough implementations of a hypertext system (Berners-Lee and Fischetti 1999, 50), which made linking information together easy. As such, WWW-based discussion platforms did not only provide a context for the communication that took place on that platform, but also provided a context for that platform as a whole by linking it to other sites within the WWW.

Contemporary CMC increasingly often takes place on what is commonly known as "social network sites" or in short SNSs. These SNSs, which nowadays usually take the form of a WWW-based application with extensive features for sharing content on the one hand, and viewing a specific user's contributions and personal information on the other hand, have enjoyed an increase in popularity especially in the first decades of the 21st century. While other, older forms of CMC such as e-mail remain popular, SNSs have been a hot topic of both popular and academic writing recently because of their meteoric rise to popularity and their increasing involvement in the daily life of their users.

1.2 Social network sites – definition and typology

Social networks may thus be taken as a subset of CMC technologies, but given how broad CMC is as a category, this designation still asks for a more specific definition. One of the earliest and probably still the most influential definition of social networks, or more specifically Social Network Sites (SNSs) was proposed by danah boyd and Nicole Ellison in their 2008 text *Social Network Sites: Definition, History, and Scholarship*. This article is an attempt to establish a foundation for further research on the topic of social networks and as such

proposes a definition of the term and a short overview of earlier work on the topic. It therefore presents a useful point of departure for an attempt to establish how social networks fit within the broader category of computer-mediated communication.

After an analysis of platforms often considered to be social network sites, boyd and Ellison arrive at defining SNSs as:

> [W]eb-based services that allow individuals to (1) construct a public or semi-public profile within a bounded system, (2) articulate a list of other users with whom they share a connection, and (3) view and traverse their list of connections and those made by others within the system (2008, 211)

In other words, an emphasis is placed on establishing an identity within the social network site, and utilizing the "web" of connected identities as a navigation tool to explore the social network site. This web of connections has often been called the "social graph", after the concept used within sociology to map conventional social networks (boyd and Ellison 2013, 153).

While this definition does fit well with most sites often considered to be social networks, such as Facebook, Twitter or MySpace, it is so broad that it also includes platforms not often thought to be a part of this category. YouTube, for example, allows its users to create profiles and "follow" the updates of certain other users, which establishes a social graph much in the sense boyd and Ellison describe. On the other hand, social interaction is not the main purpose of the platform; uploading and watching videos is. Many sites with a focus

other than social interaction, such as Flickr[6] (a photo repository), Wikipedia[7] (a collaboratively edited encyclopedia) and GitHub[8] (a programming code repository), still offer methods to establish "connections" to other users, often by way of tracking their uploads or contributions. This would make them as much of a social network as Facebook or Bebo, by boyd and Ellison's definition, while they clearly have a different focus than those sites.

Boyd and Ellison do partially address this issue by distinguishing between social network sites and social networking sites. The latter label is taken to imply that "relationship initiation", or establishing new connections, is a core activity on the site, while otherwise the focus will rather be on maintaining and cultivating existing social relationships (211). However, they do not address the question whether it is relevant to have social connections — either establishing or maintaining them — as a *primary* feature, which would still put GitHub and Facebook in the same boat.

This perceived overbroadness of the definition is one of the main issues with boyd and Ellison's work that other authors have pointed out. Most prominently, David Beer has argued that the mere distinction between social network and social networking sites is too simplistic, and that given the fact that there is a wide variety of what kind of social features sites offer and what kind of emphasis they place on their usage, a more sophisticated typology is required (Beer 2008, 518).

Such a typology has been proposed by Mike Thelwall in his 2009 text *Social Network Sites: Uses and Users*. In response to, amongst others, Beer's criticism,

---

6 http://www.flickr.com

7 http://www.wikipedia.org

8 http://www.github.com

Thelwall distinguishes three categories of social network sites (23), based on the different ways social network features are employed by these sites:

- Socialising SNSs, where the primary function of social network features is "recreational social communication"; this includes activities such as chatting, sharing links, and other low-intensity and low-stakes social interaction. Examples of such sites would be Facebook, Baidu and VKontakte.

- Networking SNSs, which includes what boyd and Ellison described as social networking sites; forging new social ties and establishing new connections is the primary focus. Networks like LinkedIn and online dating sites would be good examples.

- Navigation SNSs, where the main purpose of the site is discovering new content and information, and the social network embedded within the site is a means to do so. YouTube would be, as described before, an example of this; the social features of this site are primarily intended to allow users to discover new content.

This typology allows for a clear distinction between, for example, YouTube and Facebook; as mentioned above, they firmly fit within separate categories. Importantly, Thelwall does note that it can be a user's way of using a site that decides within what category a site fits in his or her case, rather than the site's own intention (24). For example, Twitter may be used for "recreational social communication", which would make it a Socialising SNS, but one could also follow another user's account just to see what kind of links to other sites are shared on its timeline, which would rather place it in the realm of Navigation SNSs.

This inclusion of user agency as a factor is important, as it indicates that it is not only a platform's features that define what kind of social network site it is – as could be inferred from boyd and Ellison – but also how these features are employed by the platform's users. It emphasizes that a platform's place within a user's social context is not only dependent on the platform's features and affordances, but also on how the user utilizes these.

Still, most popular contemporary social network sites would appear to fall within the Socialising SNS category. Facebook, which in March 2013 announced that it had 1,15 billion monthly active users (Facebook Investor Relations 2013, n.p.), is clearly focused on recreational social communication; likewise, VKontakte, Odnoklassniki and Baidu[9], also boast hundreds of millions of users combined (Rusbase 2013; TechInAsia 2013). While some Navigation SNS sites such as YouTube are also immensely popular, visitors often do not use the social features of the site but rather discover the content it offers via other means. For instance, many videos achieve popularity through aggregate sites like Reddit or Digg, or the recommendation algorithm YouTube uses to automatically find videos a user is likely to appreciate (Zhou et al. 2010, 409).

1.3 Before SixDegrees: Social Network Technology Ecosystems and Foucault's Genealogy

All in all, boyd and Ellison's definition of Social Network Sites, when coupled with Thelwall's extension of a more detailed typology and the addition of user agency as a factor, offers a useful way to categorize and classify the contemporary ecosystem of Social Network Sites. Ecosystem is here taken

9 VKontakte, Odnoklassniki and Baidu are Facebook-like networks that serve the Russian-speaking, Russian and Chinese market, respectively. Many other similar (but smaller) networks exist, often focused on a specific country; examples are Cyworld (South Korea), Orkut (India and Brazil) and mixi (Japan).

to mean both the SNS itself, as well as the social, political and technological context in which the SNS exists; in other words, not just the social network, but also the technological network and the power network, and how these all influence each other. Communication platforms like SNSs never exist in isolation, and they are shaped by their context while simultaneously affecting this context itself, through the way they mediate interpersonal communication. This interplay has been explored by authors such as Ulises Mejias, who wrote that communication technology is “neither a neutral tool nor an autonomous agent” (Mejias 2001, 211) and that “humans and technology shape each other” (212). Acknowledging this, it is impossible to make a meaningful analysis of the development of CMC technologies and by extension Social Network Sites without taking this context into account; user agency nor a platform’s technological, social and political “heritage” can be ignored.

However, boyd and Ellison’s overview does not go far beyond the contemporary, which presents a problem when one attempts to analyze social network sites comprehensively and as a whole rather than only the subset of recently created ones. Boyd and Ellison present SixDegrees as "the first recognizable social network site" in their analysis (2008, 214). SixDegrees was founded in 1997 and, accordingly 1997 is taken as a starting point of online social networks; as discussed earlier, criticism of boyd and Ellison's work primarily focuses on deficits in its proposed definition, not on the type of sites proposed as fitting the "social network site" label.

SixDegrees, however, did not spring into existence overnight. As boyd and Ellison themselves note when discussing how SixDegrees' features fit their definition, "each of these features existed in some form before SixDegrees, of course" (2008, 214). Yet the way in which these features, their adoption and their implementations evolved to the point where it made sense to combine them into what would become a blueprint for the recent, immensely popular

phenomenon of social network sites remains largely unexplored. The work that has been done often focuses on the content that is produced using these technologies, and the nature of the social interactions they afford, rather than the technologies themselves (see Herring 2001, 111; Turkle 2011; boyd 2008a; Castells 2001; Kiesler 1997, x). There seems, therefore, to be a gap in academic writing between dealing with online social interaction as "computer-mediated communication" with its roots in the very early internet on the one hand, and the modern phenomenon of "social network sites" on the other hand. While of course there are good reasons for limiting the scope of an article like *Social Network Sites*, the fact remains that while a clear academic narrative of the developments of social network sites since 1997 exists, the years before that are somewhat foggy, at least regarding the connection to contemporary social network sites.

As the accepted definition of SNSs is based on technologies from 1997 onwards, however, it is necessary to expand this definition when dealing with technologies from before this time. None of these easily fit within Thelwall's typology, which only makes sense as Thelwall's work is in turn based on boyd and Ellison, who explicitly note 1997 as a turning point. As noted, they also mention that all features that define Social Network Sites existed in some form before that, just not in combination with each other; consequently, the “forerunners” of SNSs can be concluded to implement one or some of these features. This kind of technologies – which will be called *Social Network Technologies* to distinguish from the narrower category of Social Network Sites – is therefore what this thesis will be investigating. These SNTs implement some of the features mentioned as essential to SNSs (user profiles, connections between users, and articulation of a list of these connections), but predate SNSs and as such by definition do not implement all of these features.

As argued earlier, certain features of contemporary social network sites can be traced back to older Social Network Technologies, or SNTs. Moreover, many SNTs are radically different from contemporary platforms in some aspects while being remarkably similar in other ways; for example, the Finger protocol (Harrenstien 1977) is an early example of a technology for retrieving "user profiles" and updates of people within a certain group or organization, but it can be characterized as a distributed system[10], whereas most contemporary social network sites run off a centralized server or network of servers. This shows that the current SNS "blueprint" is by no means the only possible implementation of the social features most internet users are now familiar with. This merits further exploration, which would both shed light on the genesis of this modern "blueprint" and highlight possible alternatives for future consideration.The objective of this thesis is therefore genealogically invest the history of social network sites — and technologies — roughly from the inception of the internet until 1997, when the "modern era" of SNSs as described by boyd and Ellison and Thelwall began. The concept of a genealogy is not without historical baggage; most prominently, Nietzsche and especially Foucault have written about the moral and epistemological genalogy at length. It is useful to consider Foucault's work on this subject for a moment to establish what kind of genealogy precisely this thesis deals with.

Foucault, drawing on Nietzsche's work, considered genealogy to be a methodology for disseminating ontological origins of knowledge. Foucault

10 The distinction between distributed, decentralized and centralized networks is a distinction between the different ways a network can physically be structured. In a centralized network, all network clients connect to one central server, which contains all data; in a decentralized network, there are several central servers which can all be connected to by clients to retrieve data; in a distributed network there is no distinction between servers and clients, as all nodes in the network are connected to each other and through each other. This typology was pioneered by Paul Baran in his early work on how to make a telecommunications network as robust as possible (Baran 1964).

distinguishes amongst others *Ursprung*, *Herkunft* and *Entstehung*[11] (1977, 77); here the *Ursprung* is the origin, the very source of a concept or artifact (78), while *Herkunft* denotes the lineage, the place within a certain group or — in the societal sense — tribe or race (81). The *Entstehung* finally is the "emergence"; the "moment of arising" as a result of an "interaction", a "struggle" and, in the Foucauldian context of power relations, "the hazardous play of dominations" (83). Contrasting this to the other two conceptions of a genealogy it is clear that the first two are rather more essentialist than the *Entstehung*, which denotes a process rather than a singular defining moment of conception in time, space or culture. Similarly, as the *Entstehung* is a process in which multiple factors mutually influence each other and in turn influence the outcome through this interaction, this emergence cannot be traced back to one singular point of origin; rather, the process is the origin.

It is, therefore, the *Entstehung* of social network sites that this thesis explores. The goal is not to sketch a linear, causal chain of technologies and systems that directly lead to the social graph-based SNSs that dominate the contemporary internet. Instead, it attempts to describe the various ways the problem of computer-mediated communication in the internet era has been dealt with, and how these may have influenced each other, and as such shaped the ecosystem in which current SNSs exist. This is particularly relevant in the case of alternative social networks, which have only found limited success in their project of disrupting the dominance of powerhouses like Facebook and Twitter. In part, it has been argued, due to their focus on re-implementing the template these SNSs have set forth rather than radically reinventing the SNS itself (Hui and Halpin 2012, 110). This thesis attempts to address this issue by both outlining alternative – relative to contemporary practice — approaches

11 These German terms are borrowed unchanged from Nietzche's *On the Genealogy of Morality* (1887).

to online social interaction, as seen in earlier forms of social network technologies, and describing what kind of ecosystem these technologies were developed in. The latter is essential as, as has been noted, such technologies are always a product of this ecosystem; what works in one context may not work in the same way in another context. Additionally, this method takes user agency in shaping the way the technology is used is into account.

As such, this thesis takes a genealogical approach to describing the history of social network theories, As in most of his writing, in his analysis of genealogy Foucault focuses heavily on power relations. This approach could also have merit with regards to social network sites — which especially in recent years have become a major factor in political debate (cf. Morozov 2011, 156; Lovink 2012, 10). However, though power relations and other contextual factors will be considered in this thesis — as noted before, all are an essential part of the *ecosystem* in which SNTs exist and as such an analysis of an SNT cannot do without them — the starting point of the analysis will be the kind of features the SNTs offer to their users. A different feature set or focus has often been how one SNT initially distinguishes it from the others; additionally, the definitions and analyses of SNSs on which this thesis builds take a feature-oriented approach as well. As such, an overview of social network technologies past and present, in which their features and, crucially, the why of these features will be discussed is the next step in this analysis.

2 An overview of the ecosystem(s)

The definition of social network technologies proposed in the previous chapter is, by necessity, quite broad. An exhaustive overview of all relevant technologies and systems would therefore be unwieldily expansive. I will instead first discuss the development philosophies and attitudes towards software design, with regards to SNTs, that were prevalent through the decades. As this obviously had a great influence on the social network technologies that were developed during these times, this generalized overview presents a good foundation for a more specific discussion of aspects in which older and newer SNTs are often different.

As for older SNTs, there are three technologies I will mostly focus on in my analysis: IRC, a protocol for group chat (fig. 2); USENET, a newsgroup system; and Finger, a protocol for retrieving personal information about people within a specific group (fig. 3). IRC, USENET and Finger have in common that they all incorporate one or more of the features that boyd and Ellison have presented as being essential to social network sites. Furthermore, IRC and USENET (and their various predecessors and derivatives) have been two of the most popular SNTs during the early decades of the internet and still enjoy popularity, which makes them suitable for comparison to contemporary SNTs. Finger has largely fallen out of use, but presented features often thought essential to contemporary SNSs and as such makes for a great "ancient" counterpart to modern social network paradigms.

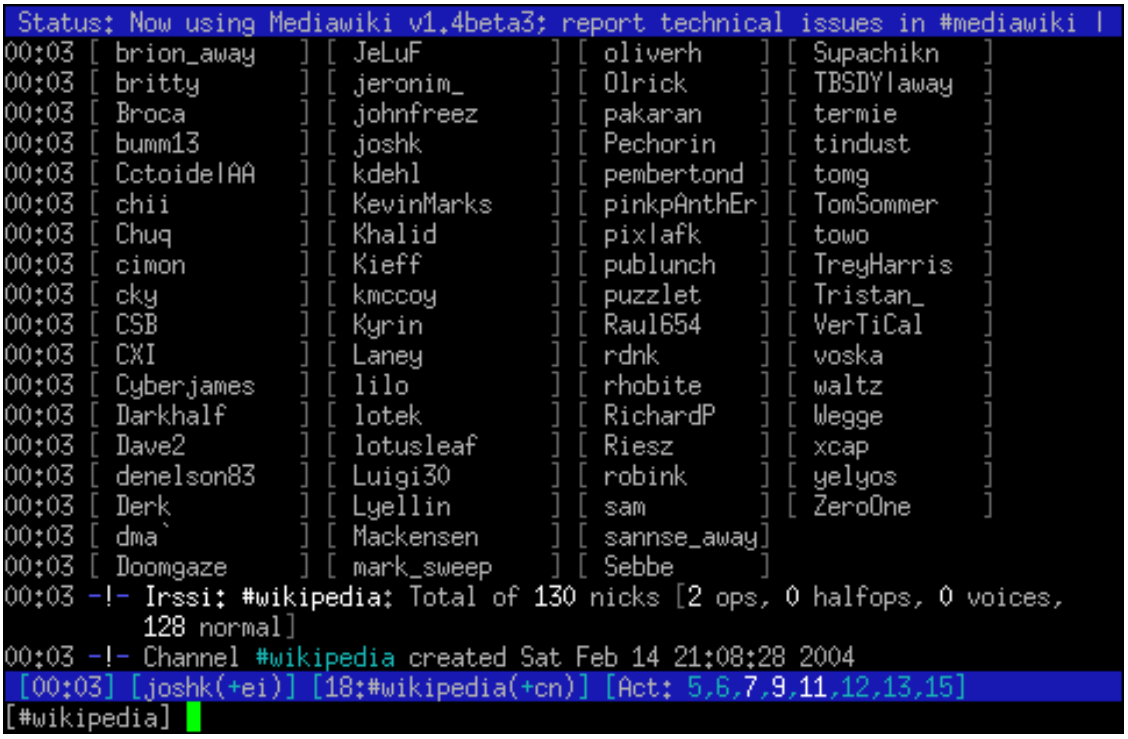

Fig. 2 — List of people currently present in an IRC channel, seen in the *irssi* IRC client.

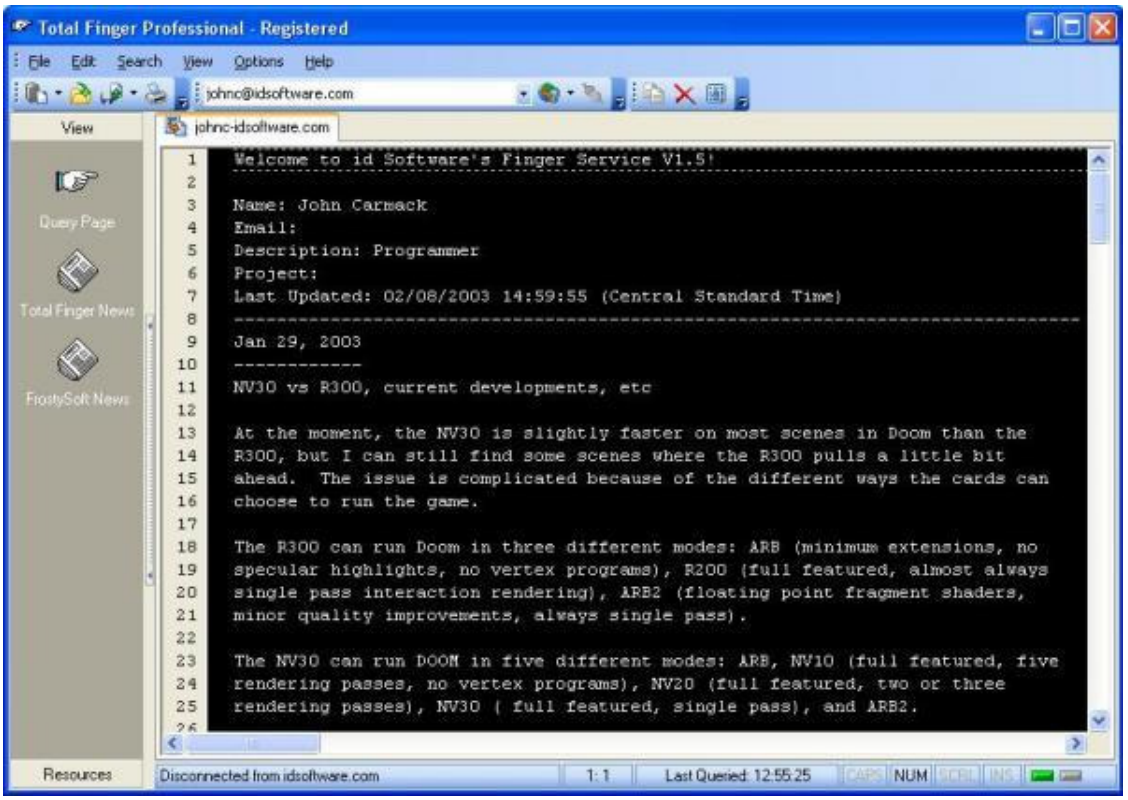

Fig. 3 — Result of a Finger query, seen in the *Total Finger Professional* client.

These three technologies were all developed before the 1990s — USENET and Finger in the late 1970s (Ryan 2011, 80; Hafner and Lyon 1996, 216), IRC in 1989 (Oikarinen n.d.; n.p.; Oikarinen 1993, n.p.)[12]. As such they are clearly from a different "era" than contemporary technologies, which makes them suitable for a genealogical investigation. Given their coming from another era, the technological and political context in which they were developed was quite different from today's as well. I will discuss a few software development paradigms that were dominant back then but may no longer be around in the same form today, so as to paint a more comprehensive picture of the environment that gave rise to these older technologies, before taking a closer look at some of the key aspects in which older and newer SNTs differ.

2.1 Small is Beautiful: The Unix philosophy and early software development paradigms

A great example of the programming paradigms during the early days of computing is the Unix operating system. Unix, which was the most popular operating system during the first decades of internet usage (Ryan 2010, 108; Naughton 1999, 306), had a distinct development philosophy. This philosophy was articulated by Doug McIlroy, one of the main original contributors to Unix, as follows:

> Write programs that do one thing and do it well. Write programs to work together. Write programs to handle text streams, because that is a universal interface. (quoted in Raymond 2003, n.p.)

This philosophy is reflected in several early CMC technologies. For example, the original program to send e-mail was called MAIL; to read received e-mail,

12 Though IRC was in turn based on older, similar chat protocols like BITNET Relay Chat and MUT (Multi User Talk); see Oikarinen n.d., n.p.

however, one would have to start the separate MAILBOX program (Hafner and Lyon 1996, 190; Ritter 2007, n.p.). Likewise, through the Finger protocol a server could show an overview of what people were connected to a server and how to reach them; to actually send them a message, however, a separate program like MAIL would have to be used.

This design philosophy could be seen as a contributing factor to why, in the words of boyd and Ellison, "each of these [SNSs] features existed in some form before SixDegrees" (2007, 214). The attitude of most Unix users – and by extension, internet users – at the time was that combining these features into one package was not needed even if it was possible — by separating features into different software packages, every single one could be good at one specific task, and people could swap them out for an alternative if they wanted.

This philosophy was sustainable for a long time, as the typical internet user was, in the 1970s and 1980s, often a researcher or otherwise computer-savvy. Dealing with command line interfaces and knowing what program to use for what purpose was how they'd learned using computers (Walsh and Bayma 1997, 385). As computer and internet usage become more widespread, however, and Unix was replaced as the most popular operating system by graphical platforms such as Windows, OS/2 or MacOS. The modular, atomized approach advocated by the Unix philosophy gave way to more integrated, multi-purpose software that allowed users to, for example, send and read mail with the same program.

SNTs evolved along roughly parallel lines. Instead of having to look up a user's details and whether they were online using Finger, then trying to set up a TALK session with them, new instant messaging software consolidated all these tasks in one user-friendly package. As software incorporated more and

more features, people would often grow to be dependent on one particular software package rather than a modular set of programs.

Another characteristic of software and technology development during the early years of the internet was its spirit of openness and sharing. The internet started as a project to let researchers throughout the United States share research data more efficiently: instead of having to fly print-outs of experiment results across the country, the information could be sent from coast to coast within a few seconds (Hafner and Lyon 1996, 41; 44). While the project was funded by ARPA, the research arm of the American Department of Defense, the web was mostly used for communication between academic staff and other people who managed to get on the network (189). There was a great sense of community; people freely shared new programs they had written to use the internet in novel ways, and rarely denied requests from others to see the program's source code and extend its functionality.

This open ecosystem fostered quick innovation and evolution of software and made sure that potentially anyone could benefit from others' work. As the amount of people using the internet was still relatively small and most of those people knew their way around a computer, anyone could join in discussions about new protocols and contribute to forming a consensus that would decide on how these standards would be set in stone through a rather informal process (Naughton 1999 35; Galloway 2004, 122; 133; Hafner and Lyon 1996, 144).

The availability of open, thoroughly discussed and widely adopted standards, coupled with the "small is beautiful" philosophy this meant that it was relatively easy to write new software that was usable in conjunction with other software implementing the same protocol. New software was released quickly

and often (Ryan 2010, 52) and its source code was usually freely shared with other enthusiasts (cf. Hafner and Lyon 1996, 207)[13].

These two components – an attitude towards development akin to the Unix philosophy, and free sharing of programs' source codes, lead to an ecosystem in which many technologies for social interaction between people online were proposed, implemented, and adopted, or in some cases rejected, with the collaborative ecosystem often making for open technologies that were developed on a basis of consensus amongst potential users, as those were often all more or less involved due to the small size of the early internet. The next chapter will take a closer look at some of these technologies.

2.2 Key aspects of social network technologies

The ecosystem in which computer technologies are developed has clearly changed since the early days of the internet; though open source and free software still has its place, many areas of computing have been changed by the commercial interests of companies like Microsoft, Facebook or Apple. When looking at the differences and similarities between the various social network technologies that have been around through the years, there are a few aspects in which there are obvious differences between these SNTs as well. These aspects are, as most design decisions are, a product of the ecosystem in which their respective technologies were designed. It is especially this ecosystem – which shapes a technology's *Entstehung* – that is interesting, as an analysis of it would shed light on why these technologies are like they are, and in some cases also why they are not like they used to be any more; for example, technological constraints may have limited communication systems up to a

13 This attitude was perhaps most succinctly encapsulated in word and deed by the Twitter message Tim Berners-Lee, one of the inventors of the World Wide Web, sent out from stage during the opening ceremony of the 2012 Olympic Games: "This is for everyone". https://twitter.com/timberners_lee/status/228960085672599552

certain breakthrough in computer capabilities, or there may have been little incentive to develop user-friendly interfaces when most computer users were accustomed to command-line interfaces. I will highlight five key aspects in which some technologies clearly differ from others, and attempt to expose how their respective ecosystems have shaped them.

2.2.1 Persistent versus volatile interaction

One of the most prominent features of contemporary social network sites is the “timeline”. The timeline often allow users to view another user's contributions even when these were made weeks, months or even years ago, in a comprehensive and chronological overview. Information is retained in such a way that both the users themselves and their connections who visit their profile page can easily retrieve updates from way back. The most prominent examples of this are modern SNSs, but the concept is not new; for example, the ancient Finger protocol allowed users to "finger" another user and retrieve their .plan file, which could contain anything but usually took the form of a chronological list of whatever the user had been working on recently (Zimmerman 1991, n.p.).

In contrast, other technologies may discard received messages and other content after one use session. This is often the case with chat-like technologies in which the chat server only functions as a relay that forwards messages from one user to another. In this case there is no central place to store information and retrieve it later. It is often possible to store received messages locally, but this is optional and always carries the risk of losing the archive in a computer crash or due to carelessness.

At first glance, such technologies may seem to be outside the realm of SNTs as they often do not offer the possibility of creating persistent user profiles or traversing another user's contact list. However, in practice this is not

always the case. Looking at IRC — Internet Relay Chat, a chat protocol for "group conferencing" developed in 1987 — the picture is less clear. IRC does indeed not support storing messages on its servers or creating persistent profiles (Oikarinen and Reed 1993, n.p.). However, it does in a way support "articulating a list of users with whom they share a connection" (boyd and Ellison 2007, 211), as IRC is fundamentally based around the concept of "chat channels", labelled chat rooms, in which users engage in conversation with the others that have joined that room. As these channels are often focused on a specific topic, offer a list of users within that channel, and allow anyone to look up the list of other channels a user has joined, this makes it possible to gain a quick overview of what kind of people have joined a particular channel and what kind of interests a specific user has.

Yet, IRC does not offer the possibility to get an overview of a user's personal details, recent content he or she has put online, or an archive of contributions. IRC was intended to be a protocol solely for internet chat and as such never included features for anything but volatile interaction between groups of users, or “text based conferencing” as the protocol’s authors call it (Oikarinen and Reed 1993, n.p.). On the other hand, this relative simplicity gave the users a great degree of freedom and agency to use the protocol as a foundation for more advanced applications. Consequently, IRC users have taken matters in own hands and created many tools to add features to IRC that were never anticipated nor developed by the protocol's creators. Many of these user-added features took the guise of a "bot", or chat robot, a computer-controlled client that connected to an IRC server, joined a chat channel and then monitored the conversations in that channel while responding to commands given by users. As these bots were often online at all times, they could be relied on as a source of information on what had happened in the channel even when the user requesting the information had been absent. For example, the popular

Eggdrop IRC bot[14] has features such as looking up the last time another user had said anything within the channel, claiming a specific nickname within the network (whereas nicknames are distributed on a first-come first-serve basis by default) and, indeed, storing personal information for a specific user.

As such, even a CMC technology that was originally designed for a "volatile" system in which no data was stored persistently can be extended with third-party features that give it characteristics which place it firmly within the realm of SNTs. A key difference, of course, is that in this case the decision of which features will be implemented and supported can be made on a per-network basis and, more importantly, by the users themselves; on most IRC networks, any user can bring a bot into a channel which can then be configured and tweaked at will. This is a fundamental difference with most persistent SNTs, such as the contemporary Social Network Sites which are usually tightly controlled by one central entity and offer users little to no influence over what kind of features are offered. The difference between volatile and non-volatile SNTs can, then, be mitigated by users as long as the ecosystem gives the users the freedom to build on top of the existing technology; IRC's simplicity and lack of central oversight gives users great agency in extending the protocol and using it as they see fit, while contemporary SNSs rarely allow users such freedom.

2.2.2 Implicit versus explicit social connections

Besides the timeline, or lack thereof, another staple feature of contemporary SNSs is the act of "friending" or "following" a user. This friending comes in two flavours; on some sites confirmation is needed from both users before a mutual connection is established, while on other sites the connection is one-way and no confirmation is needed. What these methods of establishing a connection

14 See http://www.eggheads.org/

have in common is that they are explicit: the connections are recorded in the SNS's database and often articulated in a friend list or timeline. The earliest form of such friend lists can be found in chat clients such as PowWow or ICQ (Forsberg et al. 1998, 4), though the first SNT to allow other users to view and navigate friend lists was social network site SixDegrees, in 1997 (boyd and Ellison 2007, 214).

On the other hand, some platforms and technologies do not utilise such an explicit social graph and don't maintain lists of connections between users. Many "forum"-like technologies, such as USENET, BBSs mailing lists and online discussion boards do not even offer their users the option to establish a connection with other users. Rather, interaction is centred around discussion threads, where replies to posts are grouped together so a clear line of arguments can be distinguished. Additionally, users may establish a general kind of profile about themselves by posting a message introducing themselves and having personalized signatures below their messages. This reflects the technological heritage of these systems; USENET for example was clearly developed as a mail-like application where users that knew each other would send messages around; it proved to be far more popular than that, and groups grew larger than anticipated (Naughton 1999, 178; Ryan 2010, 81; cf. Lewis and Knowles 1997, 221; Kollock and Smith 1996, 115), but the mail-like origins shine through in these characteristics.

This affords a certain kind of *implicit* connection between users. As users will gravitate towards threads concerning topics they are interested in, and avoid topics they do not care about, users will often end up being part of a "core group" that dominates certain kinds of discussions (Baym 1995, 156; Baym 1998, 56). As such, they will learn to look out for each other in other threads as well, and thanks to their extended interaction know what kind of person they are dealing with. Furthermore, thanks to signatures and introductory posts,

users are recognizable and share some amount of personal information with others, which can be used by those others to judge whether their posts are worth reading (Baym 1995, 155).

A similar but slightly more explicit form of connections can be seen in IRC channels. As each channel provides a list of users in that channel, and users are notified if another user joins or leaves, people within the channel can at all times know who else can read their conversation, even if they do not join the discussion themselves (Oikarinen and Reed 1993, n.p.). Thus, connections between users are implicitly formed by being in the same channel, and as such explicitly articulated by the list of people within one channel; additionally, when someone joins a channel all current occupants are notified of this (Rintel and Pittam 1997, 528). However, being in the same channel does not necessarily mean that there is a meaningful connection between people, especially not in large channels where the number of users may simply be too high for everyone to know each other. In such cases the web of meaningful connections will be implicitly formed in the discussion-board kind of way; through sustained conversation (cf. Rintel and Pittam 1997, 518).

Elsewhere, discussion boards have implemented features that aid forming such implicit connections – and make them more explicit again – by allowing people to "rate" a post or contribution, giving it a positive or negative score based on how they think it contributes to a discussion. This changes the dynamics of discussion and the way social relations are formed and maintained, as often posts with a higher score or by a poster with a higher aggregate score are displayed much more prominently than those with lower scores. As a result threads are dominated by posts with high scores rather than by prolific posters, and users gain renown based on the perceived quality of their posts rather than their veterancy.

Such a system based on score, or "karma" as it is often called, can be implemented in several ways. Technology news site Slashdot, which was one of the earliest prominent adopters of a karma system, displays posts in chronological order but simply hides those that have been given a low score, in an attempt to cope with the large amounts of comments news posts typically received (Ganley and Lampe 2006, 149). Popular social news site Reddit, on the other hand, which has similar problems with large amounts of user contributions, sorts posts by an algorithm that is largely based on how much karma a post has, and puts low-scoring posts near the bottom of the page and hides them by default. As a result, fame (and infamy) on Reddit is mainly achieved by gathering karma, and there are multiple examples of people adopting a certain posting style just to gather as much karma as possible (Bergstrom 2011, n.p.). As such, "implicit connections" are no longer formed on basis of sustained interaction or shared interests, but guided by the explicit karma system, which may distort these relations through its influence on posting style and perceived value as an objective marker of post quality (ibid.); while the user of course still has the option to judge posts by themselves, the ecosystem in which Reddit discourse takes place firmly guides this; the social network on this platform is shaped by the network as it exists in a technological sense.

There are similarities between karma-based systems and contemporary SNS's Facebook's "like"-button, where one can show approval of a post or comment by clicking a button and posts show a counter displaying the amount of "likes" they have a acquired. This would hint at a sort of continuum with wholly implicit systems like USENET on the one hand and Facebook on the other hand, with Reddit in the middle. However, Facebook does not change the way it displays a friend list or its time line based on the amount of likes a user has

acquired[15]; additionally, content from anyone not on the friend list is typically not shown; as discussed in chapter 1, these networks are often not focused on forging new connections and mostly facilitate maintaining existing friendships. On the other hand, sites and services with implicit connections often show content from anyone; as such, they facilitate establishing new connections. This is a fundamental difference, as a system that relies on implicit connections has no obvious way of hiding content by those not considered to be a connection by the user.

Forums, new groups and mailing lists often allow users to include their contact information in a message (either via dedicated input fields or using a general-purpose signature). Consequently, when one has established an implicit connection on such a platform a next step could be to establish an explicit connection as well using one of the alternative ways of contacting the user in question. Before the advent of social network sites this was often done via e-mail or, later, via instant messaging: it was customary to include an ICQ number or AOL nickname in signatures. Now SNSs focused primarily at maintaining existing connections have become popular, these may be the logical next step after getting to know someone in a looser fashion.

2.2.3 Synchronous versus asynchronous communication

One of the main aspects in which SNTs may differ is synchronicity. Truly synchronous SNTs are relatively rare; in the case of textual communication, which is often the case, it requires messages to be sent character-by-character, while line-by-line is more common. However, one of the earliest messaging systems in the 1970s, the Unix TALK command, offered a split-screen chat

15 Facebook does by default rank stories based to a classified algorithm; while "likes" probably play a role in this, the algorithm is classified and also takes post time into account. Additionally, the order can easily be changed to be fully chronological.

environment in which each side of the screen updated in real-time with what the user that side represented was typing (Herring 2005, 119). While chat client ICQ offered a similar option in the 1990s (ibid.), most instant messaging (such as on IRC or Facebook, or within a MUD) is only "near-synchronous", in that it waits for the user to press the "send" button before sending a line[16].

How "near-synchronous" these systems are does not only depend on the technological limitations and possibilities offered by the respective protocol, but also on the social context in which it is used. While in some contexts it might be acceptable to use short lines or even fragments of lines, other, often more formal contexts may demand putting several lines of text within one message, at which point communication becomes similar to e-mail. E-mail is clearly an asynchronous technology, as it is usually used for sending larger, self-contained messages (Herring 2005, 114). While this is to an extent convention, the fact that e-mail always requires an intermediate mail server for sending and receiving mail — it is not possible to send mail directly to the recipient — shows that even at its core e-mail is not intended or optimal for direct, synchronous communication[17].

With regards to SNSs though, there has been a marked shift from asynchronous to near-synchronous communication on platforms like Facebook and Twitter lately. While at their inception most of these sites required a

16 A prominent example of truly synchronous CMC would be voice chat, or VoIP. The earliest examples of this date back to 1995, but adoption was – and to some extent still is – hampered by the high bandwidth and hardware requirements and comparative ease of use of text chat (see Blake 2007).

17 Some chat technologies (including IRC) do also require the server as a relay for messages, at least by default. A major difference however is that these technologies are typically push-based (messages are immediately sent to the recipient upon arriving at the intermediate server) while e-mail was conceived as pull-based (messages have to be explicitly requested from the server before they are delivered).

periodic refresh to see if there were any new messages or updates, nowadays both sites — and many others — show a notice only seconds after a new message is received. As mentioned, there are several examples of near-synchronous or even truly synchronous messaging in a SNT context decades before SNSs became popular, so that raises the question why this feature was seemingly ignored first and then only implemented later. While it cannot be ruled out that the creators of Facebook and Twitter simply didn't see semi-live updates as a useful feature, a feasible alternate explanation is that the platform of these SNSs — the world wide web — was not suited to such (near-)synchronous features yet; while the social network could potentially have benefited from it, the technological network did not support such functionality.

E-mail, IRC, ICQ and MUDs, as well as Unix TALK and similar technologies, are all specialized protocols that were created expressly with whichever mode of synchronicity they required in mind. The World Wide Web runs on the HyperText Transfer Protocol (HTTP), which is a versatile general-purpose protocol, but in principle not suited for continuous synchronous communication; rather its aim is to facilitate serving different types of content on demand (Berners-Lee and Fischetti 1999, 44). HTTP is a stateless protocol, which means that the server with which a client communicates does by default not retain any "memory" of earlier interactions with a client, and was designed for request-response interaction, meaning that in order to receive information a client has to explicitly request it (Berners-Lee et al. 1999, n.p.). These characteristics are at odds with (near-)synchronous chat, in which chat messages would ideally be "pushed" to the recipient on submission and in which the server needs to keep track of who is chatting with who; as such applications relying on HTTP, which includes WWW-based sites, face technical limitations with regards to implementing any synchronous or near-synchronous applications (Bozdag et al. 2007, 15).

Only around the year 2000 near-synchronous applications on the WWW became feasible, as the XMLHttpRequest technology – an implementation of the HTTP protocol that allowed web developers to query a server without reloading a web page – was supported by a wider variety of browsers and developers; originally implemented by Microsoft's popular browser Internet Explorer, it remained an obscure technology for years until it was rediscovered as a way to address the increasing demand for real-time "Web 2.0" applications (O'Reilly 2007, 35; Woolston 2007, 8; van Kesteren 2012, n.p.). As both client-side and server-side technology evolved to a point where "push" communication (where the server sends the information without request) and sustained chat sessions became feasible even on the large scales SNSs worked with, near-synchronous communication made something of a comeback on these networks. Facebook now offers a built-in instant messaging service, and Twitter notifies users of new tweets within seconds of them being sent. Users can likewise immediately reply to messages, which brings communication on these platforms closer to the existing practice of IM clients and IRC.

Whereas many forms of CMC were (near-)synchronous during the first few decades of the internet's existence, asynchronous communication became more prevalent with the advent of the WWW as a platform for social network sites. This forced SNSs, as far as they were web-based, into a more mail-like mode of communication, where new messages would only be received upon refreshing a page. As technology progressed, near-synchronous communication became viable within the WWW ecosystem as well, and hence many SNSs have adopted this to some extent to allow users to have more immediate interactions with others. Given this trend, and the continued popularity of various forms of instant messaging, it seems likely that this (re-)convergence towards (near-)synchronous communication on social network services will continue; meanwhile, various instant messaging services and SNTs like IRC,

which predate contemporary SNSs, present specialised platforms that already offer users sophisticated (near-)synchronous communication features.

2.2.4 Transparent versus opaque software platforms

Another distinction that can be applied to SNTs is whether they are open or closed. This distinction works on both the software level (does a platform use closed or open source software?) and a policy level (is whatever entity is governing the system open about the way they handle users' data and activities?) As such it is perhaps better to speak of a distinction between "transparent" and "opaque", as "open" and "closed" are often taken as referring to whether a piece of software's source code is publicly available or not in computer-related contexts.

At first sight it would seem that while modern-day SNSs are often rather opaque about their operations — using proprietary software, and often being vague about how exactly their users' data is stored and handled — the first decades of internet were a valhalla of openness and transparency. Indeed, as has been often noted, the internet was created as a platform for easily sharing software and research data and was used as such by many. Source code to software was shared freely, fostering a climate in which software was often quickly improved on by enthusiastic users (cf. Herring 1996, 207; Naughton 1999, 138; King et al. 1997, 12).

This seems a far cry from today, where the software platforms large SNSs on are usually closely guarded corporate secrets and the networks routinely reach headlines with privacy scandals. This is a simplified portrayal of reality — one of the first source code disputes was over the very software that controlled the ARPANET (Hafner and Lyon 1996, 233) — but at least with regards to most older SNTs it does largely hold true. Before considering the implications of a

shift from transparent to opaque platforms, it is therefore useful to analyze how this ecosystem shift came to be in the first place.

Looking at SNTs like USENET, BBSs and IRC, what they have in common is that they are all protocol-based. In other words, the words "USENET", "BBS" and "IRC" refer to a specific protocol, a set of rules and regulations that describe how software packages implementing the protocol should communicate with each other. This means that anyone could develop an IRC client, a USENET reader or a BBS server, and as long as they implemented the protocol correctly anyone else could connect to them, provided their software implemented the protocol correctly as well (see Lessig 2006, 47; Galloway 2004, 121-122). Because the protocols were usually public — many were developed in the early days of the internet when this was the standard modus operandi — and the result of a consensus-based development process (see Hafner and Lyon 1996, 247), in principle anyone could implement them.

On the other hand, later software — prominent early examples are instant messengers like ICQ and AIM — often used proprietary protocols, of which details were not made public (Jennings et al 2006, 16). As such, they had a de facto monopoly over software implementing the protocol, ergo their own messaging software. This had several advantages for the protocol's designers — they could sell the software, or monetize it in other ways, and generate revenue this way. Additionally, they not only controlled the client software but also the server software, which meant they were the only ones who could host a server and as such all traffic over the messaging network went through them, which allowed for targeted advertising and further revenue-generating business models.

While it is possible to reverse-engineer a protocol, analysing traffic between client and server and trying to figure out the "language" they use to speak

to each other, this has several major drawbacks. The most obvious one is that the monopolist — ICQ or AOL — could change the protocol at any time without prior notification, and any software relying on reverse-engineering would become defunct (Jennings et al. 2006, 16). In general, service will be less reliable and the monopolist's implementations will be usually be the most popular (17). In the other case, where the protocol is freely available, people may write their own client software, adding new features to the platform – which may well influence the social interaction itself – and allowing the platform to flourish even after the original developer no longer supports it.

Given the financial advantages, it is not surprising that later SNTs often opted for such proprietary, undisclosed protocols. This had the aforementioned side-effect of giving them the uncontested monopoly over client software for their social network platforms, though many platforms offer limited third party access through APIs[18]. As a result, users can no longer easily develop their own software for interfacing with the network, in contrast to for example IRC where the protocol is open, relatively simple and stable, allowing for an extremely wide choice in client software and, by extension, client software features (see also Peeters 2013).

The shift from transparent to opaque has parallels with the shift from the internet being a relatively small network focused on researchers and other "early adopters" to a worldwide network where corporate interests play a major role. As stakes have become higher, and prospects of profit and loss likewise, an opaque platform has become more attractive than a transparent one for many starting SNS companies. On the other hand, transparent

18 Application Programming Interfaces, which allow third-party developers to contact to a service on that service's terms. For example the Twitter 1.1 API, https://dev.twitter.com/; the Facebook Graph API, https://developers.facebook.com/docs/reference/api/ or the YouTube API, https://developers.google.com/youtube/.

platforms have a much larger longevity — while SNSs like SixDegrees and Friendster closed down once they were no longer profitable, decade-old SNTs like IRC and USENET are still used by hundreds of thousands of users worldwide without any central entity paying the bills or controlling the networks (cf. Gelhausen 2013, n.p.; Galloway 2004, 119). This can at least in part be attributed to the fact that because of USENET and IRC's open nature, they were far more adaptable to a changing ecosystem than closed, centrally managed networks like many early SNSs.

2.2.5 Centralized versus decentralized system architecture

As mentioned in the previous chapter, decade-old SNTs like IRC still enjoy widespread popularity, partly thanks to their open nature. Another factor is that they do not rely on one central, controlling entity to manage network traffic and keep the platform online. Whereas most post-1990s networks are kept in firm control of whatever company owns its servers, IRC and USENET traffic is spread through hundreds or even thousands of servers which are all controlled by different entities (Galloway 2004, 120; Minar and Hedlund 2001, 5). Likewise, the Finger protocol was designed with the idea that each company or server would implement its own response mechanism (Harrenstien 1977, n.p.). Users do not connect to "the IRC network", "Finger" or "USENET" but to a specific IRC network, USENET server or Finger server, and then have access to content and user data on that network only.

While these separate networks may themselves still be centralized — meaning that all traffic on the network goes through one central node in the network — the "IRC" and "USENET" ecosystems as a whole are decentralized, in the sense that taking down one of the servers or networks will not be the end of the whole system. On the other hand, taking down Facebook's server will mean

the end of Facebook, and — as demonstrated during, for example, the few hours following Michael Jackson's death — the same goes for Twitter.

The practical implications of this decentralization vary per SNT. For IRC, a channel might move to another server when a large majority of its users have done likewise, or the original server is no longer suitable for their needs. For USENET, servers often synchronize with each other, so it would in many cases be possible to move between servers without any noticeable difference. Finger, finally, works on the premise that each organization or server only discloses information about the people on that server; data is not redundantly spread through multiple servers and servers typically are not interconnected, so one would need to know which server a user is connected to before requesting his status or .plan file.

It is no coincidence that these technologies are both transparent (see 2.2.3) and — to some extent — decentralized. Being based on an open protocol, and having freely available server software, these platforms essentially allow anyone with an internet connection to start their own server or network. While any server with non-trivial amounts of traffic will require a high-capacity computer and connection, in principle it is anyone's game. On the other hand, people cannot start their own Facebook server. This seems to be a large factor in the longevity of USENET, IRC and Finger; even in the latter's case, where few servers still support the protocol, it is re-appropriated every so often in initiatives like Thimbl[19] and Webfinger[20]. While there are drawbacks, such as people being spread through different servers without an easy way to connect

19 An initiative to use the Finger protocol to create a decentralized Twitter-like microblogging system: see http://www.thimbl.net/

20 A modernized version of the Finger protocol, aimed at usage over the Web via the HTTP protocol rather than as its own separate system: see http://tools.ietf.org/html/draft-ietf-appsawg-webfinger-17

to each other, these could also be seen as strengths, giving users the option to choose exactly with which people to connect.

A decentralized architecture can therefore be considered to be an important factor in a platforms longevity. On the other hand, it rules out a monopoly over the network, which would be financially unattractive for many companies and means that the development of the platform cannot be controlled centrally. Furthermore, in some cases it would require the users to find out for themselves what network within the SNT ecosystem to connect to, and how, which could be a turn-off for less tech-savvy users; this was less of a concern when most of the internet's user base was in fact tech-savvy, but has become an issue as internet adoption increased. There are therefore clear reasons for why currently popular networks are centralized and have one point of entry. On the other hand, it is also clear that precisely because older networks were decentralized, they still exist and enjoy some amount of success.

2.3 Conclusion

As has been discussed, there are several aspects in which older Social Network Technologies and newer Social Network Sites differ from each other. At first glance it would appear that for most of the aspects discussed in this chapter, older SNTS fall on one end of the scale while SNSs occupy the other side; contemporary SNSs have been shown to often be centralized and based on explicit social relations, for example, while for older technologies the opposite is often the case.

It would however be a mistake to construe this as one of the ends of the scale being better or worse than the other. As has been discussed, why a certain technology ended up on one end of the scale is often the result of a process in which the technology's development was steered one way or the other by a network of contextual factors such as the financial climate at the time or

what kind of technology is available. In other words the *Entstehung* of SNTs included many factors that may no longer be the same. Effecting a meaningful change on contemporary SNTs is therefore not simply a matter of adopting one of the paradigms of older SNTs as discussed in this chapter; the technology may be the same, but the ecosystem isn't, and what made sense back then may not do so now. Instead, a closer investigation is required to see whether the aforementioned aspects of older SNTs could be meaningful in addressing contemporary issues with social network sites, which is what the next chapter will be concerned with.

3 Contemporary issues with social network sites

In the previous chapter, several key aspects in which older and newer social network technologies may differ were identified. It has also become clear that these differences are not always the result of conscious choices for another paradigm, but were a product of changing circumstances with regards to internet usage, corporate climate or technological possibilities.

Being as prominent in online life as they are, contemporary SNSs have come under increasingly intense scrutiny as they grow their influence on their users' daily lives. News media routinely report on the way large social networks deal with their users' personal data and when a network announces changes to its privacy policy, user backlash is often substantial (for example, Facebook's privacy policy changes have elicited tens of thousands of angry responses on multiple occasions; see boyd 2008b, 13; Debatin et al. 2009, 85). Additionally, criticism has been levelled at the way modern social network sites seem to focus on one particular kind of online communication — what has earlier been described as "recreational social interaction" with an already-established group of friends (Thelwall 2007, 23; see chapter 1.2).

Given these issues, and the differences between older and newer SNTs, it seems worthwhile to see whether perhaps some paradigms found with older SNTs can be of use for solving these contemporary issues. While it is unlikely that directly adopting features or characteristics from older systems by current

social network sites is feasible, taking a step back and seeing whether similar issues were present in the discussed older SNTs, and how they were dealt with, could yet be beneficial to the development of improved modern SNSs; if not for mainstream sites, then perhaps for smaller, more specialized alternative SNSs.

As such, I will discuss three key issues with contemporary SNSs which have received widespread coverage in either academic or popular writing. After a short discussion of the different facets of these issues I will explore how they have been dealt with by contemporary platforms and analyze how they fit in with the findings from chapter 2, after which a general conclusion on how the issues could be dealt with can be drawn.

3.1 Privacy and data protection

Perhaps the most prominent issue with contemporary social network sites is the way they handle their user's personal data. Large networks, most prominently Facebook, have been accused of "selling user data" and tracking users against their will, following their browsing habits through their ubiquitously embedded widgets and using this data to construct detailed personal profiles of their users (Debatin et al. 2009, 83-84).

These networks often rely on selling advertisement space for revenue, and the ability to precisely target advertisements — based on a user's behaviour and the personal information they submitted to the site — can be a major selling point. While this also means that these companies probably do not "sell user data" — it would remove their competitive advantage of having the ability to target advertisements precisely — it does mean that a user's behaviour is thoroughly analysed and profiled, which makes some users uncomfortable all the same (cf. Sevignani 2013, 325-326).

SNSs gain the ability to perform this thorough profiling through tracking a user as they use their service — be it on the actual facebook.com or twitter.com website or on other sites implementing Facebook and Twitter widgets. This data is gathered on the Facebook servers, and as these are all centralized and connected together they essentially function as a giant repository of user data of all people using the service (Hui and Halpin 2013, 109; Warnke 2013 86-87).

This is to some extent a fundamental issue — if a service is designed to offer a uniform, centralized service that is more or less the same for all users, it is hard to avoid having all data on the network accessible through one central entity. As discussed in chapter 2.2.4, the alternative — a decentralized structure — often means that one can no longer connect to "Facebook" but would have to connect to one of the many Facebook servers, and consequently have access to only the data on that server.

There have been several initiatives that seek to address this issue of centralization and the inherent "data hoarding". The most prominent of these is probably Diaspora[21], a project founded by 5 ex-NYU students that raised $200.000 through crowdfunding. Eventually, this produced a software platform that has many of the staple features of modern SNSs — such as sharing photos and keeping a personal "status" — but operates through a decentralized network architecture, where anyone can start a "pod" or server. These pods can still retrieve public information about users elsewhere on the network, but do in principle not save this data, as it will quickly become out of date.

21 The project's main website is https://diasporafoundation.org/, though the network itself is, of course, spread through multiple independent sites.

While this approach alleviates some of the issues with monolithic, centralized SNS architectures, there are still serious problems with regards to safety of user data. It has been pointed out that in such a case, the system is only as strong as the weakest link: even if a user marks his or her data as private or "only available to friends", a malicious node in the network through which the data passes could ignore these settings and save the data anyway (Narayanan et al. 2012, 2). On its project website[22], Diaspora claims that "Diaspora doesn't use your data for any purpose other than allowing you to connect and share with others", and similar claims can be found on other decentralized social networks, but the fact that the network is decentralized means that this policy cannot be enforced and relies on the goodwill of the owners of the various servers that constitute the network.

This is to some extent an unavoidable problem. The "weakest link" analogy applies to any form of communication - once a piece of information is shared with someone, it is only as private as that other person keeps it. The same applies to computer communication, where in the case of SNSs it is amplified due to the fact that these services are focused on connecting to other people and looking up their information it (ibid.).

This implies that one solution to keeping data more private is limiting the amount of people with access to it. This can of course be done by encrypting it, or locking it away behind a password. That is however not an option if a network relies on "implicit" connections (see chapter 2.2.3) and limits the possibilities for exploring the network and meeting new people: it forces a network into the mould of being focused on existing relations between people, to some extent. It would seem that one cannot have the proverbial cake and eat it too; either access to people is limited to a specific network of trust, but

22 https://diasporafoundation.org/

the profiles of those people are accessible, or everything is locked away behind encryption and cannot be seen unless permission is given. This was less of a problem during the conception of technologies like USENET or Finger, when the internet was still more or less a backwater, but now commercial interests have permeated the ecosystem an open, implicit social network technology presents a set of privacy problems that seem hard to solve.

However, another factor that makes it more likely that data could be compromised is saving this data in the first place, like contemporary SNSs typically do. Volatile, (near-)synchronous platforms such as IRC do not, by default, store the conversations that pass between users; while users may log those conversations they are a part of, it is not possible for them to keep track of everything happening on the network. Likewise, USENET servers usually have a specific "retention rate", a time limit after which messages are deleted. While someone could keep archives of a server simply by connecting every so often and archiving the new messages, the premise is still that data is not kept indefinitely. As there is no commercial incentive for most servers to do so either, this premise often does hold true. Coupled with the fact that USENET and IRC activity is spread over multiple independent networks, and no explicit personal information like age, real name or gender is stored as such, this makes for platforms that are at least superficially more private and anonymous than contemporary, centralized SNSs.

3.2 Lack of user influence on platform features

Another issue that is often raised concerning contemporary SNSs is the lack of customizability. What features are offered by networks is decided by these networks themselves and while many offer the possibility to upload a profile picture or user-specific page background, usually customization beyond that is not possible. The "user experience" is tightly controlled and there is little room for configuration.

An example is the "gender" field many social networks offer on their users' profile. Usually the choice for this field is limited to "male", "female", and often not entering a value is not an option. While this is sufficient for some, there may be reasons for someone to wish not to disclose their gender, or they may not feel comfortable within either of the options offered. Yet there is no option to even choose "other" on Facebook and Google+; the service not only dictates what kind of information can be shared, but also in what form (McNicol 2013, 208).

The aforementioned "alternative social network" Diaspora handles the specific issue of gender by simply making it an open question; a user can leave the "gender" field empty or type in whatever gender designation they are most comfortable with (213). Yet even Diaspora still moulds a user profile to a certain template by offering a specific set of fields for the user to fill in. For example, there is a "gender" field, but other identifying information does not get a dedicated profile field, instead being delegated to a general-purpose 'bio' section. Furthermore, the user is still reliant on the Diaspora software platform's developers for new features or new profile fields[23].

Looking back at the older SNTs discussed in chapter 2, these technologies avoided this issue in various ways. First, since they were largely protocol-based, anyone with sufficient knowledge of the involved technology could in principle write a new client or server with the kind of features they wanted, as long as it implemented the protocol correctly. Second, in the case of user profiles, usually no explicit user profile was created at all; user profiling

23 Though, as Diaspora is an open source project, potentially anyone with sufficient programming knowledge could program a new version of Diaspora with different options, and nominate it for inclusion in the main software package. This is still far from a sure-fire way of effecting change, however, as the change would still need to be accepted by a committee of other developers; see Peeters 2013 for a more thorough discussion of these issues.

relied on inferred information and fully user-specified data such as USENET signatures and Finger .plan files, which did not use fixed profile fields apart for a few fields of automatically generated data such as the date the user was last logged in (Harrenstien 1977, n.p.).

Large SNSs are not protocol-based for various reasons that make it unlikely that this will change in the near future; it makes user profiling harder and it opens the platform up to user-made client, both of which are likely to decrease revenue gained from the service. Smaller or alternative networks however, which often do not have direct profit as a goal and instead put more focus on their user's interests, could benefit from especially the approach that IRC takes with regards to customizability.

The IRC protocol itself is rather bare-bones; it facilitates channel-based communication between users and little else. To add features to IRC chat and channels, users therefore quickly started creating "bots" that facilitated all kinds of functionality not offered by the IRC protocol itself. Bots, computer-controlled IRC clients, offer features like quizzes, relaying a message to a user that's currently not connected to the network, keeping track of when a user was last seen online and storing short bits of personal information about users that can be retrieved later. Due to the simplicity of the IRC protocol, bots are relatively easy to create — many beginners' tutorials for programming describe how to make one — and can yet offer a lot of functionality — they are only limited by the fact that IRC offers text chat as its only mode of communication.

How does this relate to contemporary SNSs? Some of these platforms do in fact offer a degree of extensibility; the Facebook Apps platform offers thousands of apps and is in principle open to anyone with a Facebook account, as long as they use the APIs Facebook provides and adhere to its guidelines.

These last two regulations, however, mean that users can still not extend the platform on their own terms, but only within a small space tightly regulated by Facebook. This can once again be traced back to commercial interests; Facebook is financially invested in its platform and is looking to make as much revenue from it as possible, in which case tight control over what is possible and what's not is beneficial. In the case of IRC, where there is no central entity paying the bills or overseeing operations, this is not a concern and as such there was no incentive to limit possibilities for user extensions.

3.3 One-sided focus on one-to-one communication

A third common criticism with regards to modern SNSs, mostly from academic circles, is the emphasis they place on the individual as a starting point for the social graph – a practice that has been said to "[be] the anti-thesis of community" (Mejias 2001, 214) and "destroy [the social] at the same time as it formalizes it" (Stiegler 2013, 22). This "atomized" representation of interpersonal relations often maps to pre-existing relationships in the "real world" or at least outside the social network. All these relationships are homogenized into one uniform concept of "friendship", which often translates to being allowed to see the contributions of such a friend to the SNS.

This dominant conceptualization of online friendship has been criticized by writers such as French philosopher Bernard Stiegler as being overly simplistic, as it formalizes and exposes friendship, while friendship fundamentally aims to "escape formalization and publicity" (Stiegler 2013, 21). Looking back at Thelwall's typology, this criticism seems to be primarily aimed at Socialising SNSs, where existent off-line friendships are often assimilated into the uniformity of the social network site; the other two types (Networking SNSs and Navigational SNSs), are on the contrary explicitly focused on discovering new people and content.

Yet, the most successful modern SNSs are of the Socialising type, so the criticism remains relevant. Authors like Harry Halpin and Yuk Hui have pointed out that the focus on homogeneous, one-dimensional "friendship" inhibits the social potential of such networks, or as they put it, "the spontaneity and innovation within their possible collective intelligence is deformed by the control of the networks, driven as it is by intensive marketing and consumerism aimed at individuals rather than the development of the potential of the group" (Hui and Halpin 2012, 107).

The premise here seems to be that because SNSs have a significant (financial) interest in focusing on individuals rather than groups, they have no reason to develop features and systems enabling groups of people to collaborate effectively, while there is a lot of potential yet to be unlocked in that area. So is there anything in older SNTs that could foster the "collective individuation", as Hui and Halpin summarize their "spontaneity and innovation" of the group (ibid.)?

It is clear that especially IRC is built around the concept of the group as the primary social unit. The focus on "channels" rather than one-on-one conversations means that virtually any IRC conversation will be a group conversation rather than a dialogue. A similar thing can be said for mailing lists or USENET, which was based on the mailing list concept; messages are by default sent to anyone subscribed to the thread at hand.

However, as described in chapter 2, these SNTs are relatively simple platforms that rely on user-made extensions or implicit actions for anything more complex than getting a message to all recipients and putting that message in a specific context (be it a thread or a channel). While this extensibility and implicitness is, as has been argued, a strength in some cases, it does at the same time mean that beyond making the channel or thread the primary

conversational context there are few, if any, directly apparent benefits with regards to the collective individuation Hui and Halpin advocate.

Yet this simple shift in focus entails a significant shift in how social interaction takes place on IRC or USENET. The implicit nature of forming relationships on these platform — as opposed to the formalized process of "friending" that is currently in vogue — has often shown remarkable strength in fostering close-knit communities (Baym 1998, 32; Reid 1991; Turkle 2011, 58). The lack of formality and freedom in deciding how many personal information is shared means that users can communicate on their own terms. Coupled with a context that virtually always includes more than two people and — in the case of IRC — near real-time conversation speed, this makes for an environment in which close collective bonds are easily formed (and broken).

Still, a strong platform for forming communities is not the same as a strong platform for collective individuation, which is mobilizing these communities to successfully collaborate on producing content (Hui and Halpin 2012, 114). Both USENET and IRC chat are volatile and often closer to chit-chat than constructive brainstorming. Furthermore, however anonymous the users within these communities can choose to be, any communication still clearly originates with one user rather than the collective. True collaborative content production tools such as Instant Update[24], MoonEdit[25] or Etherpad[26] provide

24 Instant Update was an application for the original Macintosh, released in 1991 by ON Technology, that allowed people to work on one document together, with updates to the text appearing on all computers in real-time.

25 MoonEdit is a Windows application released by Tom Dobrowlski in 2005, with similar collaborative editing options and also a feature to scroll back and forth through the editing history in real-time. See also: http://moonedit.com/

26 Etherpad is much like MoonEdit and Instant Update, but runs in a web browser rather than as a separate application. See also: http://etherpad.org/

a more stable platform for actual collaboration, but usually forego any social features that could bring people together to start working on things in the first place.

Older SNTs, through their reliance on implicit social relations and group-focused modes of interaction, clearly offer a fertile ground for forming new communities, and as such do have something to offer with regards to effecting a shift from a focus on the individual to the collective as a primary unit of social networks through these features. This on its own, however, is only a part of what Hui and Halpin described as "collective individuation", true social collaboration that produces content originating with a group rather than an individual. Tools for such collaboration exist, but these often do not offer their users many possibilities for establishing or maintaining connections with collaborators. There is therefore no clear-cut solution here — but it is clear that a group-focused platform focused on implicit social relations could at least offer a solid foundation for collective individuation.

3.4 Conclusion

There are several aspects contemporary social network sites that have been criticized by both users and scholars. Concerns with regards to privacy have received widespread attention, but criticism has also been aimed at the lack of user customizability and focus on the uniform, individual user as the base node in the social network on these platforms.

Some paradigms or features found in older SNTs could address these concerns to some extent. Privacy issues would to a certain degree be addressed by adopting a decentralized architecture like many older protocols mandate, though this would not mitigate all privacy risks; privacy was a lesser concern back when internet was less ubiquitous, and as such the ecosystem in which technologies like USENET, IRC or Finger were developed did not stimulate

systems that specifically address these issues. User customization on the other hand is made relatively easy by bare-bones, open protocols like IRC, though this will probably be at the cost of base functionality and ease-of-use, as it typically requires programming skills to utilize this freedom. The focus on the individual, finally, is to some extent addressed and mitigated by USENET and IRC's volatile, implicit social relations, which are not as homogenous as the rigid structure most SNSs impose.

All in all there seems to be merit in taking a closer look at features of older SNTs as a possible starting point for addressing issues with contemporary SNSs, though it has been shown that some of these features and paradigms emerged as a result of a specific ecosystem that might not make them as feasible nowadays. Still, measures like adopting an open protocol or making it easy for users to customize their experience as outlined in chapters 2 and 3 would certainly be possible, at least on a smaller scale where commercial considerations play less of a role.

4 Discussion and conclusion

Computer-Mediated Communication technologies offer a rich and varied spectrum of methods to communicate with other people over the internet. These technologies have been developed since the very inception of the internet, with some dating back as far as the early ARPANET. For a long time these often simple technologies were the main mode of communication for people online. In recent decades, WWW-based platforms dubbed "social network sites" or SNSs have become the dominant platform for such communication.

These SNSs have been shown to be relatable to older technologies such as IRC, USENET and Finger in many ways, be it through implementing features similar to what these technologies offer or adopting a paradigm that is almost a direct inverse of what these forerunners did. Either way, many features of contemporary platforms can easily be compared directly to one or the other feature of older platforms, showing something of a lineage from back then to the current day. In chapter 2, it has been shown that there seems to be something of a continuum for some of these features or design choices, where the older technologies are on one end and the newer technologies on the other hand. This shows that SNTs have evolved quite radically, at least in some aspects, over the past decades.

The evolution from the way these features were implemented before to how they are adapted on modern platforms is often categorized by a desire for a more uniform representation of users and their online relations, and — related

to that — the fact that most prominent modern social network platforms are centralized, for-profit efforts; as such it is lucrative to keep tight control over what happens on these platforms. This can be contrasted with the more free and open software design culture of the early days of the internet, when the network existed in an ecosystem that made this possible; many users were intimately familiar with computer technology and commercial interest was low. As this ecosystem changed, SNTs became more tightly controlled, both in the sense that they were now often owned and designed by one central entity and in the sense that users were profiled and needed to conform to a certain kind of profile.

This approach has led to concerns about issues such as privacy, lack of influence from the users of a platform and a one-sided view of how online social interaction should work. Looking back at how older technologies dealt with this online social interaction, several aspects of these forerunners that could disrupt this status quo have been identified; they could offer alternatives, as in their *Entstehung* the factors that shaped contemporary SNSs were not present. However, this on the other side also means that what worked back then does not necessarily solve the problems the current ecosystem causes. In some aspects it could certainly effect a meaningful change, though; especially the possibilities open protocols like IRC offer to their users seem useful in solving problems like the lack of user influence on a platform's features.

It is unlikely that the current social media powerhouses will adopt any of these aspects – as they have little to gain from it, at least in the short term. As these mainstream platforms still have perhaps the most powerful argument to seduce users – all their friends are there too – SNSs that could take the risk of adopting these different paradigms will probably reach a lower amount of users. However, for Social Network Sites for which this is not necessarily a

concern – like those with a very specific membership centred around a certain topic or political ideal – there could yet be some interesting opportunities here for these platforms to, in a sense, remember where they came from — and broaden the horizon and address SNS issues in a meaningful way.

Bibliography

Baran, Paul. 1964. "On Distributed Communications Networks." *IEEE Transactions on Communications Systems* 12 (1): 1–9.

Baym, Nancy K. 1995. "The Emergence of Community in Computer-Mediated Communication." In *CyberSociety: Computer-Mediated Communication and Community*, edited by Steven G. Jones, 138–163. Thousand Oaks: SAGE Publications.

———. 1998. "The Emergence of On-Line Community." In *CyberSociety 2.0*, edited by Steven G. Jones, 35–68. Thousand Oaks: SAGE Publications.

Beer, David. 2008. "Social Network (ing) Sites… Revisiting the Story so Far: A Response to Danah Boyd and Nicole Ellison." *Journal of Computer-Mediated Communication* 13 (2): 516–529.

Bergstrom, Kelly. 2011. "'Don't Feed the Troll': Shutting down Debate About Community Expectations on Reddit.com." *First Monday* 15 (8). doi:http://dx.doi.org/10.5210%2Ffm.v16i8.3498.

Berners-Lee, Tim, R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, and P. Leach. 1999. "RFC2616: Hypertext Transfer Protocol -- HTTP/1.1". Network Working Group. http://tools.ietf.org/html/rfc2616.

Berners-Lee, Tim, and Mark Fischetti. 1999. *Weaving the Web: The Past, Present and Future of the World Wide Web by Its Inventor*. London: Orion Business Books.

Blake, Errol A. 2007. “Network Security: VoIP Security on Data Network–a Guide.” In *Proceedings of the 4th Annual Conference on Information Security Curriculum Development*, 27.

boyd, danah m. 2008a. “Taken Out of Context: American Teen Sociality in Networked Publics”. Berkeley: University of California-Berkeley, School of Information.

———. 2008b. “Facebook’s Privacy Trainwreck: Exposure, Invasion, and Social Convergence.” Convergence: *The International Journal of Research into New Media Technologies* 14 (1) (February 1): 13–20. doi:10.1177/1354856507084416.

boyd, danah m., and Nicole B. Ellison. 2007. “Social Network Sites: Definition, History, and Scholarship.” *Journal of Computer-Mediated Communication* 13 (1) (October): 210–230. doi:10.1111/j.1083-6101.2007.00393.x.

———. 2013. “Sociality through Social Network Sites.” In *The Oxford Handbook of Internet Studies*, edited by William H. Dutton. Oxford: Oxford University Press.

Bozdag, Engin, Ali Mesbah, and Arie Van Deursen. 2007. “A Comparison of Push and Pull Techniques for Ajax.” In *Web Site Evolution, 2007. WSE 2007. 9th IEEE International Workshop On*, 15–22.

Castells, Manuel. 2001. *The Internet Galaxy*. Oxford: Oxford University Press.

Custer, C. 2013. “Baidu Cloud Breaks 70 Million Users, Growing at 200k Users a Day.” *TechInAsia*. June 27. http://www.techinasia.com/baidu-cloud-breaks-70-million-users-growing-200k-users-day/.

Debatin, Bernhard, Jennette P. Lovejoy, Ann-Kathrin Horn, and Brittany N. Hughes. 2009. “Facebook and Online Privacy: Attitudes, Behaviors, and Unintended Consequences.” *Journal of Computer-Mediated Communication* 15 (1): 83–108.

“Facebook Reports Second Quarter 2013 Results.” 2013. *Facebook Investor Relations*. July 24. http://investor.fb.com/releasedetail.cfm?ReleaseID=780093.

Forsberg, Mattias, Kristina Höök, and Martin Svensson. 1998. “Design Principles for Social Navigation Tools.” In *Proceedings of the 4th ERCIM Workshop on'User Interfaces for All', Special Theme'Towards an Accessible Web', Stockholm.*

Foucault, Michel. 1984. “Nietzsche, Genealogy, History.” In *The Foucault Reader*, 76–100. London: Random House.

Galloway, Alexander. 2004. *Protocol: How Control Exists After Decentralization*. Cambridge: The MIT Press.

Ganley, Dale, and Cliff Lampe. 2006. “How Deep Runs the Karma? Structural Holes and Social Capital in an Online Community.” In *MWAIS 2006 Proceedings*, 147–153. Association for Information Systems.

Gelhausen, Andreas. 2013. “IRC Networks - Top 10.” *irc.netsplit.de*. September 11. http://irc.netsplit.de/networks/top10.php.

Hafner, Katie, and Matthew Lyon. 1996. *Where Wizards Stay Up Late: The Origins of the Internet*. New York: Simon and Schuster.

Harrenstien, Ken. 1977. “RFC742: NAME/FINGER”. Network Working Group. http://tools.ietf.org/html/rfc742.

Herring, Suzan C. 2005. “Computer-Mediated Communication on the Internet.” *Annual Review of Information Science and Technology* 36 (1): 109–169.

Hopkins, Ben. 2013. “VK.com #9 in the World, But Odnoklassniki Is Closing the Gap.” *Rusbase*. July 4. http://rusbase.com/news/author/benhopkins/vkcom-9-world-odnoklassniki-closing-gap/.

Horton, M., and R. Adams. 1987. “RFC1036: Standard for Interchange of USENET Messages”. Network Working Group. http://tools.ietf.org/html/rfc1036.

Hui, Yuk, and Harry Halpin. 2013. “Collective Individuation: The Future of the Social Web.” In *Unlike Us Reader: Social Media Monopolies and Their Alternatives*, edited by Geert Lovink and Miriam Rasch, 103–117. Amsterdam: Institute of Network Cultures.

Jennings, Raymond B., Erich M. Nahum, David P. Olshefski, Debanjan Saha, Zon-Yin Shae, and Chris Waters. 2006. “A Study of Internet Instant Messaging and Chat Protocols.” *Network, IEEE* 20 (4): 16–21.

Jones, Steven G. 1995. Preface to *CyberSociety: Computer-Mediated Communication and Community*, vii–ix. Thousand Oaks: SAGE Publications.

———. 1998. Introduction to *CyberSociety 2.0*, xi–xvii. Thousand Oaks: SAGE Publications.

Kesteren, Anne van. 2012. “XMLHttpRequest: W3C Working Draft 6 December 2012.” Edited by Julian Aubourg, 송정기, and Halvord R.M. Steen. World Wide Web Consortium.

Kiesler, Sarah. 1997. Preface to *Culture of the Internet*, ix–xvi. Mahwah: Lawrence Erlbaum Associates.

King, John, and Rebecca E. Grinter. 1997. "The Rise and Fall of Netville: The Saga of a Cyberspace." In *Culture of the Internet*, edited by Sarah Kiesler, 3–34. Mahwah: Lawrence Erlbaum Associates.

Kolko, Beth, and Elizabeth Reid. 1998. "Dissolution and Frafmentation: Problems in On-Line Communities." In *CyberSociety 2.0*, edited by Steven G. Jones. Thousand Oaks: SAGE Publications.

Kollock, Peter, and Marc Smith. 1996. "Managing the Virtual Commons: Cooperation and Conflict in Computer Communities." In *Computer-Mediated Communication: Linguistic, Social, and Cross-Cultural Perspectives*, edited by Suzan C. Herring, 109–128. Amsterdam: John Benjamin.

Lessig, Lawrence. 2006. *Code 2.0*. New York: Basic Books.

Lewis, David D., and Kimberly A. Knowles. 1997. "Threading Electronic Mail: A Preliminary Study." *Information Processing and Management* 33 (2): 209–217.

Lovink, Geert. 2012. "A World Beyond Facebook: Introduction to the Unlike Us Reader." In *Unlike Us Reader: Social Media Monopolies and Their Alternatives*, edited by Miriam Rasch and Geert Lovink. Amsterdam: Institute of Network Cultures.

McNicol, Andrew. 2013. "None of Your Business? Analyzing the Legitimacy and Effects of Gendering Social Spaces Through System Design." In *Unlike Us Reader: Social Media Monopolies and Their Alternatives*, edited by Geert Lovink and Miriam Rasch, 347–363. Amsterdam: Institute of Network Cultures.

Mejias, Ulises. 2001. “Sustainable Communicational Realities in the Age of Virtuality.” *Critical Studies in Media Communication* 18 (2) (June): 211–228. doi:10.1080/07393180128075.

Minar, Nelson, and Marc Hedlund. 2001. “A Network of Peers: Peer-toPeer Models Through the History of the Internet.” In *Peer-to-Peer: Harnessing the Power of Disruptive Technologies*, edited by Andy Oram, 3–21. Sebastopol: O’Reilly.

Morozov, Evgeny. 2011. *The Net Delusion: The Dark Side of Internet Freedom*. New York: PublicAffairs.

Narayanan, Arvind, Vincent Toubiana, Solon Barocas, Helen Nissenbaum, and Dan Boneh. 2012. “A Critical Look at Decentralized Personal Data Architectures.” *arXiv Preprint* arXiv:1202.4503. http://arxiv.org/abs/1202.4503.

Naughton, John. 1999. *A Brief History of the Future: The Origins of the Internet*. London: Weidenfeld and Nicolson.

Nietzsche, Friedrich. 1887. *On the Genealogy of Morality: A Polemic*. Leipzig.

O’Reilly, Tim. 2007. “What Is Web 2.0: Design Patterns and Business Models for the Next Generation of Software.” *Communications and Strategies* (1): 17.

Oikarinen, Jarkko. “IRC History.” *irc.org*. http://www.irc.org/history_docs/jarkko.html.

Oikarinen, Jarkko, and D. Reed. 1993. “RFC1459: Internet Relay Chat Protocol”. Network Working Group. http://tools.ietf.org/html/rfc1459.html.

Peeters, Stijn. 2013. “The IRC Paradigm: An Alternative Approach for Alternative Social Networks”. Utrecht: Universiteit Utrecht. http://stijnpeeters.nl/docs/report-irc.pdf

Raymond, Eric S. 2003. *The Art of Unix Programming*. ed 1.0. n.p. http://www.faqs.org/docs/artu/index.html.

Reid, Elizabeth M. 1991. “Electropolis: Communication and Community On Internet Relay Chat”. Melbourne: University of Melbourne. http://www.irchelp.org/irchelp/communication-research/academic/academic-reid-e-electropolis-1991.html.

Rintel, E., and Jeffery Pittam. 1997. “Strangers in a Strange Land: Interaction Management on Internet Relay Chat.” *Human Communication Research* 23 (4): 507–534.

Ritter, Gunnar. 2007. “Mail, Mail, Mailx, Nail—history Notes.” *Heirloom Project*. January 28. http://heirloom.sourceforge.net/mailx_history.html.

Ryan, Johnny. 2010. *A History of the Internet and the Digital Future*. London: Reaktion Books.

Sevignani, Sebastian. 2013. “Facebook Vs. Diaspora: A Critical Study.” In *Unlike Us Reader: Social Media Monopolies and Their Alternatives*, edited by Geert Lovink and Miriam Rasch, 323–337. Amsterdam: Institute of Network Cultures.

Steinfield, Charles W. 1986. “Computer-Mediated Communication Systems.” *Annual Review of Information Science and Technology* 21: 167–202.

Stiegler, Bernard. 2013. “The Most Precious Good in the Era of Social Technologies.” In *Unlike Us Reader: Social Media Monopolies and Their Alternatives*, edited by Geert Lovink and Miriam Rasch, 16–30. Amsterdam: Institute of Network Cultures.

Thelwall, Mike. 2009. “Social Network Sites: Users and Uses.” *Advances in Computers* 76: 19–73.

Turkle, Sherry. 2011. *Alone Together: Why We Expect More from Technology and Less from Each Other*. New York: Basic Books.

Walsh, John P., and Todd Bayma. 1997. “Computer Networks and Scientific Work.” In *Culture of the Internet*, edited by Sarah Kiesler, 385–406. Mahwah: Lawrence Erlbaum Associates.

Warnke, Martin. 2013. “Databases as Citadels in Web 2.0.” In *Unlike Us Reader: Social Media Monopolies and Their Alternatives*, edited by Geert Lovink and Miriam Rasch, 76–88. Amsterdam: Institute of Network Cultures.

Woolston, Daniel. 2006. *Pro Ajax and the Net 2.0 Platform*. New York: Apress.

Zhou, Renjie, Samamon Khemmarat, and Lixin Gao. 2010. “The Impact of YouTube Recommendation System on Video Views.” In *Proceedings of the 10th ACM SIGCOMM Conference on Internet Measurement*, 404–410. http://dl.acm.org/citation.cfm?id=1879193.

Zimmerman, D. 1991. “RFC1288: The Finger User Information Protocol”. Network Working Group. http://tools.ietf.org/html/rfc1288.