The Web as Platform

Data Flows in Social Media

Anne Helmond

The Web as Platform

Data Flows in Social Media

Academisch proefschrift

ter verkrijging van de graad van doctor
aan de Universiteit van Amsterdam
op gezag van de Rector Magnificus prof. dr. D.C. van den Boom
ten overstaan van een door het college voor promoties ingestelde commissie,
in het openbaar te verdedigen in de Agnietenkapel
op woensdag 23 september 2015, te 10:00 uur

door

Anne-Paulien Helmond
geboren te Nijmegen

Promotiecommissie

Promotor:	Prof. dr. R.A. Rogers	Universiteit van Amsterdam
Overige leden:	Prof. dr. R. Boast	Universiteit van Amsterdam
	Prof. dr. M.J.P. Deuze	Universiteit van Amsterdam
	Prof. dr. J.F.T.M. van Dijck	Universiteit van Amsterdam
	Prof. dr. Z.A. Papacharissi	University of Illinois at Chicago
	Prof. dr. E. Rutten	Universiteit van Amsterdam

Faculteit der Geesteswetenschappen

Acknowledgements

First, and foremost, I would like to thank my colleagues with the Digital Methods Initiative (DMI), the New Media & Digital Culture PhD program at the University of Amsterdam: Richard Rogers, Esther Weltevrede, Erik Borra, Carolin Gerlitz, Bernhard Rieder, Anat Ben-David, Michael Stevenson and Sabine Niederer. My promotor, Richard Rogers, has shown tremendous support and interest in my work over the years. Not only has he introduced me to a medium-specific view on the web but he also taught me how to do solid web research with such a view in mind. I am incredibly thankful to have been a founding member of DMI since the summer of 2007 when Richard invited me to work on new research methods and tools with fellow MA students Esther, Erik and Michael. DMI has been my home ever since, which as a research group is the most supportive, productive, forward-looking, energetic and innovate research environment one could wish for.

In particular I wish to thank my DMI colleagues, friends and "paranimfen" Esther Weltevrede and Carolin Gerlitz. Esther and I met during our MA degree and have worked together on numerous projects and two papers. Esther, it has been very inspiring and fun to work together and I am looking forward to our future collaborations. Carolin and I met during the DMI Summer school 2009 and we have collaborated on a number of projects over the years which have led to two co-authored papers. In addition, Carolin has provided a wonderful amount of feedback on large parts of this dissertation for which I am extremely grateful. I met Erik Borra during our MA degree and he has been invaluable for this dissertation in which I make use of a number of digital methods tools that have been written by Erik for DMI.[1] On top of that Erik has created custom-made tools, adapted to the specific research needs of chapter 3, for which I am incredibly thankful. This thesis has also benefitted greatly from feedback provided by Michael Dieter, Marc Tuters, David Nieborg and Bernhard Rieder, who also wrote me a script to follow shortened URLs (see chapter 4).

I wish to thank Geert Lovink for his continuous support and for introducing me to the field of software studies during my MA thesis and for sending me to San Diego to attend and present at the Software Studies workshop entitled SoftWhere 2008.[2] My colleague Thomas Poell has been a wonderfully encouraging roommate who was always available for dissertation questions, teaching questions and valuable career advice. José van Dijck has provided extremely valuable career advice in the final year of my dissertation and with her help I was able to further my career in academia.

[1] See the DMI tool database for an overview of tools: https://tools.digitalmethods.net [Accessed 7 April 2015].

[2] See: http://workshop.softwarestudies.com/ [Accessed 15 July 2014].

Thanks to Mathieu Jacomy for providing tools and feedback, Willem Hiddink and Robert-Reinder Nederhoed for providing a database dump of the Dutch Loglijst and Marguerite Lely and Anneke Agema for their editorial advice on chapter 3.

All the chapters of this dissertation have been presented at various conferences and I would like to thank the conference participants for their insightful questions and suggestions. In addition, I wish to thank the anonymous reviewers and editors from *Social Media & Society* (chapter 2), *First Monday* (chapter 3), *Computational Culture* (chapter 4) and *New Media & Society* (chapter 5) for their feedback and suggestions. While writing my dissertation I participated in various workshops, conferences and collaborations and have met many great people who have inspired me over the years. Here I would like to specifically thank Greg Elmer, Ganaele Langlois, Taina Bucher, Noortje Marres, David Berry and Jussi Parikka for our fruitful discussions during conferences and workshops. In the final year of my dissertation I visited the Social Media Collective at Microsoft Research and I wish to thank Kate Crawford, Nancy Baym, Mary L. Gray, Kate Miltner, Tarleton Gillespie and Jonathan Sterne for our productive discussions during my visit.

Finally, I'd like to thank the Amsterdam School for Cultural Analysis (ASCA) and in particular Eloe Kingma for her continuing support of the ASCA PhD community. My colleagues at the Mediastudies department are the nicest colleagues one could wish for. Thanks to my colleagues in the New Media & Digital Culture team as well my Film and Television studies colleagues who have supported me throughout these years, in particular Leonie Schmidt, Maryn Wilkinson and Daisy van der Zande for their encouraging words. Finally, a very warm thank you to my mom, Anneke Agema, who has always shown great interest in my work, has read and edited almost my entire dissertation and has supported me throughout. Thanks to my family for their support during these years.

Amsterdam, the Netherlands, March 2015

Acknowledgements co-authored articles

Weltevrede, Esther, and Anne Helmond. 2012. "Where Do Bloggers Blog? Platform Transitions within the Historical Dutch Blogosphere." *First Monday* 17 (2). doi:10.5210/fm.v17i2.3775.

Chapter 3 is based on a three-year collaboration (2010-2013) with Esther Weltevrede, University of Amsterdam.[3] The chapter was presented in distinct versions on several occasions over time, including Frank Meeuwsen's *Bloghelden* book launch (2010) about the history of the Dutch blogosphere in Utrecht, the Out of the Box: Building and Using Web Archives conference (2011) organized by the International Internet Preservation Consortium at the National Library of the Netherlands in The Hague and the MiT7 Unstable Platforms conference (2011) at the Massachusetts Institute for Technology in Cambridge, MA.

The empirical work for this study was carried out together in sessions in Amsterdam (2010-2013) and Barcelona (Spring 2011). The text was written collaboratively in Google Docs and is equally attributed to both authors. This chapter was rewritten to highlight the changing blogging and linking practices with the rise of social media platforms in the Dutch blogosphere.

Gerlitz, Carolin, and Anne Helmond. 2013. "The Like Economy: Social Buttons and the Data-Intensive Web." *New Media & Society* 15 (8): 1348–65. doi:10.1177/1461444812472322.

Chapter 5 is based on a three-year collaboration (2010-2013) with Carolin Gerlitz, University of Amsterdam.[4] It resulted from our work at the Digital Methods Summer School 2010 at the University of Amsterdam. During my keynote at the Summer School I asked whether the hyperlink is still the currency of the web with the emergence of new types of links characterizing Web 2.0: the subscribe, the like, the share, the number of retweets, the submit to Digg, the save to Delicious, the social network profile, the shortened URL, etc. At the end of my lecture I proposed a project to address the question whether these new link types are forming new currencies of the web.[5] This led to the 'Web Currencies' project with Vera Bekema, Carolin Gerlitz and Bretton Fosbrook and I would like to thank my collaborators for thinking through this project, which eventually resulted in the Like Economy paper co-

[3] This wiki page documents the first stages of the project in 2010: See: https://wiki.digitalmethods.net/Dmi/DutchBlogosphere2003 [Accessed 29 December 2014].

[4] When we started our collaboration Carolin Gerlitz was a PhD candidate at Goldsmiths, University of London.

[5] See: http://www.annehelmond.nl/2010/08/18/how-web-1-0-is-the-issuecrawler/ [Accessed 27 May 2014].

authored with Carolin. This paper was presented in distinct versions at several conferences over time, including the Digital Methods Winter conference (2011) in Amsterdam, the Platform Politics conference (2011) in Cambridge, UK, the MiT7 Unstable Platforms conference (2011) at the Massachusetts Institute for Technology in Cambridge, MA, the Unlike Us 2 conference (2012) in Amsterdam and the 20th International Conference on Information Science (2012) in Amsterdam. Parts of the section on 'Resisting the Like economy' were previously published in 'The Like Economy: The Politics of Data and Dataflows in the Social Web' (2012) in the conference proceedings of *BOBCATSSS 2012*, the 20th International Conference on Information Science. The Like Economy was written collaboratively in Google Docs and is equally attributed to both authors. Minor amendments were made to the version that appears as a chapter in this dissertation.

Table of Contents

List of figures

List of tables

List of tools

G-Atlas. Developed by Mathieu Jacomy for the TIC Migrations group, 2011. For more information: http://www.e-diasporas.fr/ [Accessed 1 May 2015].

Gephi. Developed by Bastian et al. (2009), the Gephi Consortium, 2008. Available at: https://gephi.github.io/ [Accessed 1 May 2015].

Harvester. Developed by Erik Borra for the Digital Methods Initiative, 2008. Available at: https://tools.digitalmethods.net/beta/harvestUrls/ [Accessed 1 May 2015].

Internet Archive Wayback Machine Link Ripper. Developed by Erik Borra for the Digital Methods Initiative, 2011. Available at: https://tools.digitalmethods.net/beta/internetArchiveWaybackMachineLinkRipper/ [Accessed 1 May 2015].

Internet Archive Wayback Network Per Year. Developed by Erik Borra for the Digital Methods Initiative, 2014. Available at: https://tools.digitalmethods.net/beta/waybackNetworkPerYear/ [Accessed 1 May 2015].

OpenRefine. Developed by David Huynh, Metaweb Technologies, Inc.[6] Available at: http://openrefine.org [Accessed 1 May 2015].

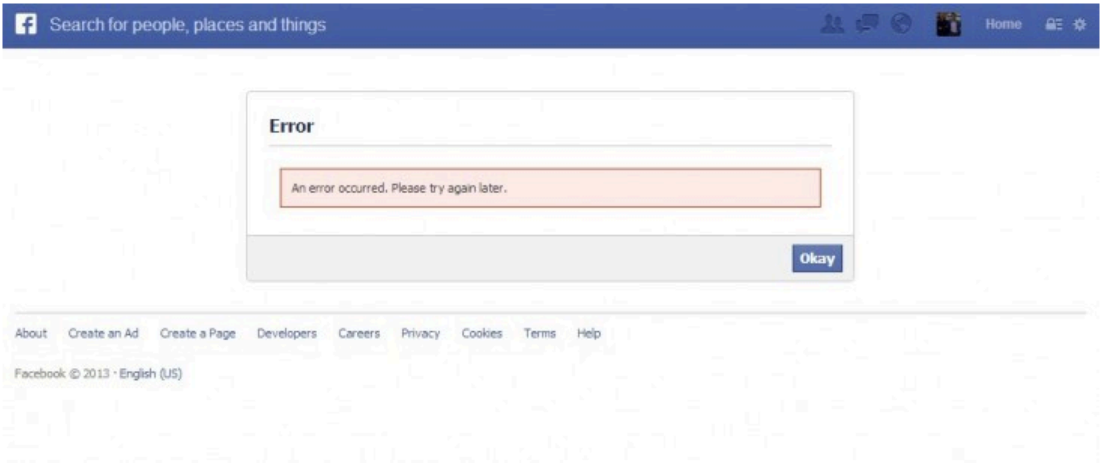
Table2Net. Developed by Mathieu Jacomy at the Sciences-Po Médialab, 2011. Available at: http://tools.medialab.sciences-po.fr/table2net/ [Accessed 1 May 2015].

Tracker Tracker. Developed by the Digital Methods Initiative, 2012. Available at: https://tools.digitalmethods.net/beta/trackerTracker/ [Accessed 1 May 2015].

URL Follow. Developed by Bernhard Rieder, 2012. Available at: http://labs.polsys.net/tools/urlfollow/ [Accessed 1 May 2015].

[6] This software was created by Metaweb Technologies, Inc. and originally written and conceived by David Huynh. Metaweb Technologies, Inc. was acquired by Google, Inc. in July 2010 and the product was renamed Google Refine. In October 2012, it was renamed OpenRefine as it transitioned to a community-supported product. See: https://github.com/OpenRefine/OpenRefine/blob/9b2a506caada4ea6d580d21140184fef8caa7566c/README.txt [Accessed 1 May 2015].

1. Introduction: Social media platforms extend into the web



Error

An error occurred. Please try again later.

Figure 1: A Facebook bug affects external websites on February 8, 2013. Image reproduced with permission from The Next Web. http://thenextweb.com/facebook/2013/02/08/apparent-issue-with-facebook-connect-is-dragging-people-from-around-the-web-to-a-moot-error-page/ [Accessed 25 March 2015].

On 8 February 2013, major websites including CNN, The Washington Post, ESPN, NBC, The Los Angeles Times, The Huffington Post, Business Insider, Slate, TechCrunch, BuzzFeed, Pinterest, Yelp and Hulu became inaccessible to many web users for about an hour (Samson 2013; Wilhelm 2013). When trying to access these websites many visitors were automatically redirected to an error page on Facebook stating: "An error occurred. Please try again later" (see figure 1). How was this possible? How did Facebook manage to break the web?

The affected websites had in common that they were integrated with the social media platform Facebook. The websites had all implemented Facebook features such as the Facebook Like button or Facebook Login by adding Facebook code to their sites. This code opens up a two-way data communication channel between Facebook and third-party websites. By adding this code, websites can connect with Facebook's platform and use Facebook functionality and content within their own pages.

On this occasion, however, the integration was severely disrupted when Facebook stopped handling this connection on their end correctly (Wayner 2013). Usually, when a visitor loads a webpage with Facebook elements, the Facebook code on the page sends information about this request to Facebook to check whether the visitor is a logged-in Facebook user (Wayner 2013). If the user is logged in, the Facebook elements on the page will be personalized. A bug in Facebook's authentication system redirected users *away* from the external websites they were trying to visit and sent them to an error page on Facebook.com instead (Wayner 2013; Wilhelm 2013).[7] This bug affected all the users who were logged into Facebook somewhere and who tried to access external websites that were integrated with Facebook.[8] With a bug in their code, Facebook "hijacked" a large number of websites, at least according to technology reporters from *ReadWrite* and *InfoWorld* (Rowinski 2013; Samson 2013).[9]

This example demonstrates two important phenomena: First, it shows how deep Facebook has become entangled with the web. It demonstrates that Facebook's features, often referred to by technology reporters as Facebook's "tentacles,"[10] reach deep into the web.

[7] Facebook's official statement was: "For a short period of time, there was a bug that redirected people from third-party sites integrated with Facebook to Facebook.com. The issue was quickly resolved." See: http://www.salon.com/2013/02/08/when_facebook_broke_the_web/ [Accessed 11 December 2014].

[8] This also affected users who were still logged into Facebook or logged into another website using Facebook Login. If you leave Facebook.com or a website with Facebook Login without actively logging out you are still logged into Facebook. Simply leaving these websites to visit another website or closing your browser does not log you out. So when logged-in users tried to go to a website that was integrated with Facebook, either through Facebook login or the ubiquitous Like button, a bug pertaining to users' logged-in state redirected them to an error page on Facebook instead.

[9] ReadWrite is one of the leading blogs about web technologies and the internet industry. Technology blogs, websites and magazines such as *ReadWrite, The Next Web, TechCrunch, Mashable, Slate, Ars Technica, VentureBeat, Gigaom* and *Wired* provide important industry commentary and analysis. In addition, individual bloggers such as software developer Dave Winer and venture capitalists such as Marc Andreessen and Fred Wilson discuss new developments on their blogs and comment on the state of the web. The trade press, technology websites and blogs function as central actors within the technology industry with early reflections on emerging web technologies, technical descriptions, reviews as well as in-depth analyses of web services and other online phenomena.
In addition, these sources also contribute to theorizing about web technologies and online culture by coining new terms such as: "the long tail" by previous *Wired* editor-in-chief Chris Anderson, "crowdsourcing" by *Wired* editors Jeff Howe and Mark Robinson, "freemium" by Jarid Lukin on venture capitalist Fred Wilson's blog post outlining the new business models of the web and "filter bubble" by *Upworthy* co-founder Eli Pariser, amongst other terms. Sometimes referred to as so-called "internet intellectuals" or "internet gurus" (see Morozov 2011 for a critical take on 'The Internet Intellectual') these authors constitute an important part of the dialogue between the industry and academia with their popular writings on new media technologies and practices.
These authors have coined multiple terms that have entered the academic discourse, also to critically interrogate said terms, illustrating the tight relationship between technology websites, blogs, the trade press and new media studies. These sources are important resources for new media scholars who are studying the web and in particular for web historians who wish to write histories of the web with a focus on particular services or technologies. They document new websites, technology companies and web technologies that may change and/or disappear over time. Similarly, they document changes of web services, such as new features or new ownership, which may not be documented or archived by the services themselves. In this dissertation these sources have been indispensable resources to analyze the developments of social media platforms over time. Technology blogs such as *ReadWrite, The Next Web, TechCrunch* and *Mashable* have observed and covered many changes to Facebook's interface and features and have extensively documented Facebook's announcements at their f8 Developers Conferences.

[10] See, for example, the following articles titles: Harbison, Niall. 2011. "Facebook Is Slowly Creeping Its Tentacles All over Your Web." *The Next Web*. May 2. http://thenextweb.com/socialmedia/2011/05/02/how-facebook-is-slowly-creeping-its-tentacles-all-over-your-web/ [Accessed 22 December 2014].
Miners, Zach. 2014. "As Facebook's Data Tentacles Spread, Privacy Questions Resurface." *PCWorld*. October 3. http://www.pcworld.com/article/2691552/as-the-tentacles-of-facebooks-data-spread-privacy-questions-resurface.html [Accessed 22 December 2014].
Richmond, Riva. 2011. "As 'Like' Buttons Spread, So Do Facebook's Tentacles." *Bits Blog*. September 27. http://bits.blogs.nytimes.com/2011/09/27/as-like-buttons-spread-so-do-facebooks-tentacles/ [Accessed 22 December 2014].

Second, it shows how these websites have become reliant on Facebook's infrastructure by integrating Facebook features. Facebook's ubiquitous infrastructure was running invisibly in the background of large parts of the web until a bug in Facebook's code caused it to malfunction. Infrastructures, so Star and Ruhleder argue, often operate in the background and only become visible when they break down (1996, 113). The breakdown also makes the complex relationship between Facebook, its platform, the web, webmasters and users visible.

The platformization of the web

The example of Facebook taking down a large number of websites due to a bug demonstrates one of the consequences of Facebook's widespread extension into the web. According to Facebook cofounder and CEO Mark Zuckerberg over 30 million apps and websites have been built using Facebook (see keynote video in Liu 2015). Website profiler BuildWith estimates that 28.4% of the top 10,000 websites have implemented Facebook features (2015).[11] However, Facebook is not the only social media platform that has been extending itself into other web spaces. In addition, platforms such as Twitter, Google+ and LinkedIn offer a number of similar features that can be integrated into third-party websites and apps: social login systems to sign into external websites and apps using existing credentials from social media services and social buttons to embed, share and recommend content from external websites and apps. Whereas only few social media platforms offer social login systems,[12] most of them offer a set of social sharing buttons. These buttons, I argue in this dissertation, are central devices for social media platforms to weave themselves into the web. To date, within new media studies, little attention has been paid to these kinds of mechanisms and their effects on the web and its users.

The extension of social media platforms into the web and its consequences—or what I refer to as the platformization of the web—is the subject of my study. I use the notion of platformization to refer to the rise of the platform as the dominant infrastructural and economic model of the social web. That is, platformization could be construed as the transition of social network *sites* into social media *platforms*. As an infrastructural model, social media platforms provide a technological framework for others to build on, which I argue, is geared towards their expansion into the rest of the web.[13] As an economic model, I put

Tate, Ryan. 2014. "Facebook's Ad Tentacles Now Infiltrate Almost Any App." *Wired UK*. May 1. http://www.wired.co.uk/news/archive/2014-05/01/f8-facebook [Accessed 22 December 2014].

[11] BuildWith scans the homepage of Quantcast Top 1,000,000 websites by traffic for implemented technologies such as content management systems, analytics and third-party widgets. 28,4% of the Quantcast Top 10,000 websites have implemented Facebook SDK for JavaScript, which sets up connections between Facebook and external websites. Quantcast is one of the leading technology companies specializing in website audience measurement.

[12] User management platform Janrain provides a list of 30 social login providers on its website which includes a number of social media platforms: https://rpxnow.com/docs/providers [Accessed 9 March 2015].

[13] In the next chapter I expand on this computational meaning of the term platform.

forward that social media platforms employ their extensions to make external web data platform ready.[14] These two processes of decentralizing platform features and recentralizing platform ready data characterize what I call the double logic of platformization. This double logic is operationalized through platform-native objects such as social buttons which connect the infrastructural model of the platform to its economic model. An example is Twitter's tweet button on external websites which automatically turns all hyperlinks shared to its platform into platform-specific shortened URLs. Twitter has repurposed the hyperlink by turning it into a data-rich shortened link that can gather data outside of its platform boundaries.

In this thesis I trace this platformization of the web and its consequences in terms of 1) the transformation of social network sites into social media platforms 2) the restructuring of the blogosphere and the introduction of new linking practices, 3) the changing nature of the hyperlink from a navigational tool into an analytical tool for data capture, 4) the transformation of the currency of the web from link to like and 5) the boundaries of a website and the end of it as a bounded object. Each of these consequences is detailed in individual case studies, where I discuss the mechanics behind the double logic of platformization. I demonstrate that the consequences of social media platforms' tight integration with the web—platformization—typify a significant change in how the web's infrastructure is put to use. For some, such as Tim Berners-Lee, the platformization of the web refers to the decline of "the open nature of the web" (1996) in relation to the rise of social media platforms (see chapter 2). That is to say, the architecture of social media platforms is geared towards connecting and integrating these platforms into other online spaces, thereby turning those spaces into instantiations of social media.

The widespread integration of social media features into websites raises a number of issues that are central to this dissertation, that is, how websites have built in social media features (e.g., widgets), and how social network websites historically have enabled their programmability by providing access to their content and functionality thereby becoming social media platforms. In chapter 2, I examine XML, widgets and APIs, and how they each enable various kinds of data exchanges, or what Alan Liu has called "data pours" (2004). In addition, I inquire into the role of the platform architecture in the decentralization of social media features into the web. In doing so, I turn to the history of the blogosphere, tracing how bloggers pioneered the utilization of sidebar widgets, thereby connecting websites not by conventional hyperlinks but rather by social media. In chapter 3, I discuss whether social media has altered the structure of the blogosphere. Here I examine the changing patterns of hyperlinking and especially how social media has encouraged different forms of linking as well as links themselves. In chapter 4, I ask, how social media has industrialized the hyperlink and introduced new link types, and to which ends. I address this question by analyzing how the social media platform Twitter handles and reconfigures hyperlinks into shortened URLs and

[14] The notion of making data 'platform ready' has been derived from Tarleton Gillespie's idea of how data is made 'algorithm ready' (2014, 168) and is further explored in chapter 2 and 4.

for which purposes. In chapter 5, I examine how social media platforms have changed the currency of the web. Here I discuss new web currencies such as the like, share and retweet. In chapter 6, I inquire into the ways in which third-party objects such as social buttons redraw the boundaries of a website. Ultimately, I turn to cross-platform analysis, and ask how we can employ these third-party objects in websites to study a website's ecosystem, and ultimately the spread of platformization and thereby its consequences.

To study the platformization of the web, I therefore argue, one should engage with data exchange mechanisms, new means to connect websites, the transformation and commodification of the hyperlink, the introduction of new web currencies for web content such likes, shares and retweets, and the redrawn boundaries of the website. That is, one should recognize the platform-specific objects that have been introduced by social media platforms which function to reweave the fabric of the web, to rephrase Tim Berners-Lee (2000).

Platform interoperability and the decentralization of platform features

The topic of this dissertation follows from two related observations on the changing nature of the web with the rise of social media platforms in the mid 2000s: first, the interoperability between social media platforms and third parties, and second, the decentralization of platform features. Here, I expand on these two observations to set up the context of the questions that are central to this dissertation.

As an avid web user since 1995,[15] I have seen the development of many new websites, services and web technologies. Within 10 years, the web as I knew it had changed significantly. This change has been framed by Tim O'Reilly as a shift from Web 1.0, the web as a medium for publishing information, to Web 2.0, the web as a medium for participation (O'Reilly 2005; Song 2010, 251). By introducing the term Web 2.0, Matthew Allen argues, O'Reilly versioned the web, similar to practices within software development to give software updates incremental version numbers to indicate new developments (2013, 264). Allen sees O'Reilly's versioning of the web as a marketing strategy allowing him "to claim [Web 2.0] to be new" and at the same time promising "an easy transition from what came before" (2013, 264). Thus, Allen contends, Web 2.0 "implies *both* continuity and change" (2013, 261) or in O'Reilly's words: "'the 2.0-ness' is not something new but rather a fuller realization of the true potential of the web platform" (2005). Following Allen (2013), in this dissertation I do not consider Web 2.0 as a neatly demarcated period of the web or position Web 2.0 as a radical break from the past (cf. Stevenson 2013). I rather trace the continuities—e.g. the use of hyperlinks to create connections—as well as the discontinuities—e.g. the introduction of new types of links—within Web 2.0.

[15] In 1994 I was introduced to the world wide web for the first time when my father brought home print-outs from The Internet Underground Music Archive and The Ultimate Band List websites. In this sense, he premediated (Grusin 2010) my first encounters with the web.

Felicia Song calls the so-called shift from Web 1.0 to Web 2.0 a "discursive move from information to participation" (2010, 252) by O'Reilly to sell the idea of Web 2.0 to the internet industry as a new post-dotcom-crash business model in which users add value to services through participation. By highlighting the alleged 'new' role of the participating user (as critiqued by Van Dijck 2009; Van Dijck and Nieborg 2009), early Web 2.0 discourses became infused with what David Beer and Felicia Song refer to as a rhetoric of "democratization", "empowerment" and "emancipation" (2009, 986; 2010, 252).[16] Central actors in this so-called "participatory web"[17] (Madden and Fox 2006; Beer 2009, 986) are websites such as YouTube, Flickr, del.icio.us, Digg, Facebook and Twitter, creating an online "participatory culture" (Benkler 2006; Jenkins 2006). These Web 2.0 services share what Tim O'Reilly refers to as an "architecture of participation" in which users add value to a site by creating and sharing content (O'Reilly 2005).[18] To gain additional data, O'Reilly recommends designing services that also "set inclusive defaults for aggregating user data and building value as a side-effect of ordinary use of the application" (2005).[19]

Thus, an important aspect of Web 2.0 services are their software infrastructures for capturing, storing, organizing and redistributing data which, O'Reilly argues, require specialized databases and "a competency in data management" (2005). He further states that within Web 2.0 applications "data is the next Intel Inside" where some categories of data function as the "building blocks for Web 2.0 applications" which are "designed for 'hackability' and 'remixability'" (O'Reilly 2005). In doing so, many Web 2.0 services make their data and functionality available to third parties through XML/RSS, widgets or APIs.[20] These mechanisms set up data channels that enable connections with third parties. In software engineering, this principle of making system connections compatible is referred to as interoperability (Geraci 1991, 217). Within the social web, Robert Bodle argues, Application Programming Interfaces (APIs) "enable interoperability or the sharing between websites and

[16] The rhetoric of democratization can be found in early utopic discourses around Web 2.0. Critiques on the so-called empowerment of users often describes participation as a form of free or immaterial labor and as a form of exploitation since value is derived from user activities (Lazzarato 1996; Terranova 2000; Coté and Pybus 2007; Petersen 2008; Fuchs 2010). Mirko Schäfer therefore suggests distinguishing between explicit and implicit participation (2011, 51). Whilst explicit participation is "driven by motivation, either intrinsic or extrinsic" implicit participation "is channeled by design, by means of easy-to-use interfaces, and the automation of user activity processes" (Schäfer 2011, 51). This complicates our understanding of participation and at the same time provides a two-fold view on how data is collected under the idea of "participation".

[17] See Michael Stevenson's critique of the web's supposed participatory turn. In a case study on HotWired, he shows how "the concepts, debates, and design aims surrounding the participatory web were very much a part of web production early on" (2014, 2).

[18] In his article entitled 'Loser Generated Content: From Participation to Exploitation' Søren Mørk Petersen argues that the architecture of social network sites and user-generated content sites such as Flickr and MySpace is geared towards the collection of user data over user content (2008). This, he argues, turns the architecture of participation into "into an architecture of exploitation and enclosure, transforming users into commodities that can be sold on the market" (Petersen 2008).

[19] This is a form of what Mirko Schäfer refers to as channeling implicit participation by software design (2011, 51).

[20] These formats and technologies that enable websites to make their data and functionality available to third parties and distribute them outside of their boundaries are the topic of the next chapter.

online services" by providing a software interface for platforms (2011, 321).[21] APIs form the underlying technological glue of the social web and allow content to circulate between platforms (Langlois, McKelvey, et al. 2009).

So the first observation that has led to the questions behind this dissertation pertains to the interoperability between social media platforms and third parties including other platforms, external websites and apps and the central role of APIs in creating these connections. The second and related observation concerns the increasing presence of platform features outside of the platform's boundaries. Here, I am referring to the use of social media sidebar widgets on blogs and social buttons on blogs and other types of websites.[22]

When I started blogging—fairly late—in September 2006,[23] I used several widgets in the sidebar of my blog. Widgets are a very popular mechanism in the blogosphere to embed content from social media platforms such as Flickr, as further addressed in chapter 2 and 3. In the Fall of 2006, I noticed a new type of widget popping up all over the web. This widget took the form of a button to bookmark or share a website to an external service such as social bookmarking website del.icio.us or social news website Reddit. These so-called social buttons function as decentralized platform features and have become the prime mechanisms to facilitate the semi-automated sharing of web content to social media platforms, as further discussed in chapter 2, 3, 4 and 5.

The observed mechanisms, platform interoperability and the decentralization of platform features, serve as the two important entry points to analyze the platformization of the web. They also challenge the early image of social media platforms as walled gardens (see chapter 2) and they have formed the conceptual basis to analyze the platformized web in ecological terms.

From walled gardens to social media platform ecologies

In the Fall of 2008, while writing my PhD proposal, I had the opportunity to explore the two observations described above by turning them into small research projects that stood at the onset of this dissertation. With our research group the Digital Methods Initiative (DMI), I attended the two-day international conference 'Walled Garden: Communities and Networks Post Web 2.0,' organized by Virtueel Platform in Amsterdam, 20-21 November 2008.[24] The

[21] APIs are further defined and discussed in chapter 2.

[22] Some initial ideas about widgets and the role of numbers in widgets to measure value and attention in the social web have been developed in: Helmond, A. (2009). Lifetracing. The Traces of a Networked Life. *H. Thorington, E. Navas, J. Green, Networked: A (networked_book) about (networked_art).* Roslindale, MA: Turbulence.

[23] I started my blog http://www.annehelmond.nl/ during the Digital Practices course as part of the MA New Media & Digital Culture at the University of Amsterdam in 2006. This course was led by Geert Lovink who inspired us to create a group blog and a personal blog to write about anything new media related during our degree. The student group blog http://mastersofmedia.hum.uva.nl/ became a big success and is still being used by the current MA students.

[24] See: http://virtueelplatform.nl/activiteiten/walled-garden/ [Accessed 12 December 2014].

Walled Garden conference focused on "the (in)accessibility of information and knowledge driven by a tendency towards online gated and closed communities."[25] The conference served as an intervention into the Web 2.0 hype by critically interrogating a new type of network that Web 2.0 introduced on the web: the social network as a walled garden. The conference website used the following Wikipedia definition to describe walled gardens as the object of inquiry:

> A walled garden, with regards to media content, refers to a closed set or exclusive set of information services provided for users (a method of creating a monopoly or securing an information system). This is in contrast to providing consumers access to the open Internet for content and e-commerce (Wikipedia 2008).[26]

This definition served as a starting point for the discussions in the working groups of the conference. With DMI, I participated in the session entitled 'Mapping the Walled Gardens: Digital Methods for Researching and Visualizing Networks on the Web,' moderated by Sabine Niederer and Richard Rogers. We started with a group discussion and addressed the term association of walled gardens (of all social networks Facebook was most associated with the term walled garden), the ontology of walled gardens, and the privacy or leakiness of walled gardens.[27]

My contribution to these discussions regarded questioning the supposed impermeable state of walled gardens. One aspect of critique is that they are closed environments that can only be entered with the right login, and in which data is locked into the network (McCown and Nelson 2009). However, social networks offer a number of features that allow data to flow in and out of the platform, which, I argued, makes them semi-permeable. In the discussion that followed we continued with the ecological metaphor to analyze the semi-permeable state of social networks as walled gardens. I described Application Programming Interfaces (APIs)[28] and widgets[29] as important features that enable these so-called walled gardens to plant their seeds outside of their gardens. In addition, I put forward that APIs offer a number of mechanisms to carefully regulate the data flows of walled gardens, turning them into tightly controlled ecosystems (see chapter 2).

In a sub-group, I continued exploring the semi-permeability state of walled gardens and put forward that the permeability of a site can be characterized by whether it provides

[25] Ibid.

[26] Ibid.

[27] See: http://www.annehelmond.nl/2008/12/22/walled-garden-communities-and-networks-post-web-20-part-1/ [Accessed 12 December 2014].

[28] An API is an Application Programming Interface, a set of functions that provides a structured exchange of data and functionality between sites and services. This term is further explained in chapter 2.

[29] A widget is a small application that can be inserted into a third-party website and display content or integrate functionality from another website. This term is further explained in chapter 2.

mechanisms for data exchanges with third parties, such as APIs,[30] embed codes, widgets, social sharing buttons and RSS feeds. We then made a number of sketches to map the data flows between different services as enabled by these data exchange mechanisms.

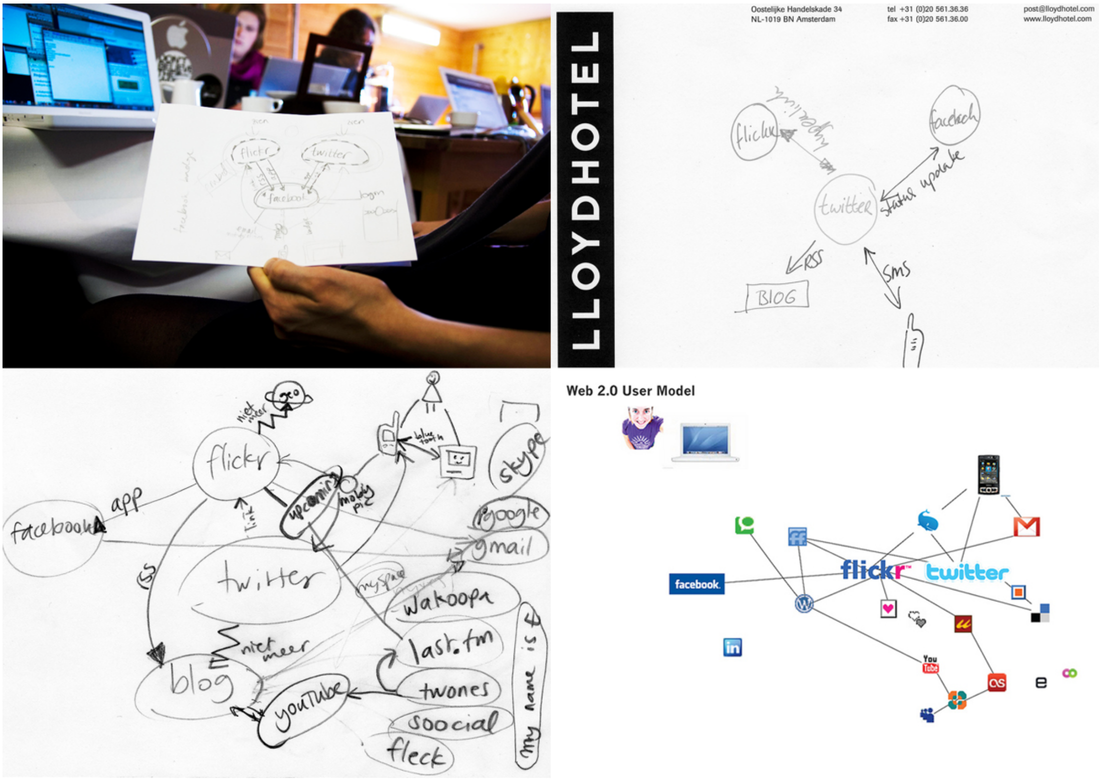


Figure 2: Four sketches showing platform interoperability among social media platforms. The sketches display the interconnections between social media platforms created by the possibility to cross-post content from one platform to another. Analysis by Anne Helmond, Sabine Niederer, Auke Touwslager, Laura van der Vlies and Esther Weltevrede. Visualizations by Anne Helmond, created with Adobe Illustrator. © Digital Methods Initiative. November 2008.

In a first step we visualized the automated connections between different social media platforms to analyze platform interoperability (see figure 2). We mapped out all the different services in use by a single web user, myself, and traced all the connections between them. We focused on the automated connections between services through cross-posting, the automatic posting of content to multiple services. For example, in 2008 I used the application Mobypicture on my mobile phone to take pictures. This app was configured to automatically cross-post these pictures to my Mobypicture, Twitter and Flickr accounts, thereby demonstrating platform interoperability. In addition, an RSS-based Flickr widget on my blog

[30] Hootsuite CEO Ryan Holmes describes APIs as the gates in the walled gardens of social networks like Twitter and Facebook in an article for FastCompany (2013). Hootsuite is an application built on top of Twitter that depends on the Twitter API.

automatically pulled in the latest five images from Flickr, creating connections between Mobypicture, Flickr and my blog.

In a second step we focused on the different types of data exchange mechanisms between three major social media platforms: Facebook, Twitter and Flickr (see figure 3). We created a visualization depicting the different devices enabling data flows between these services to analyze the extent of their permeability.

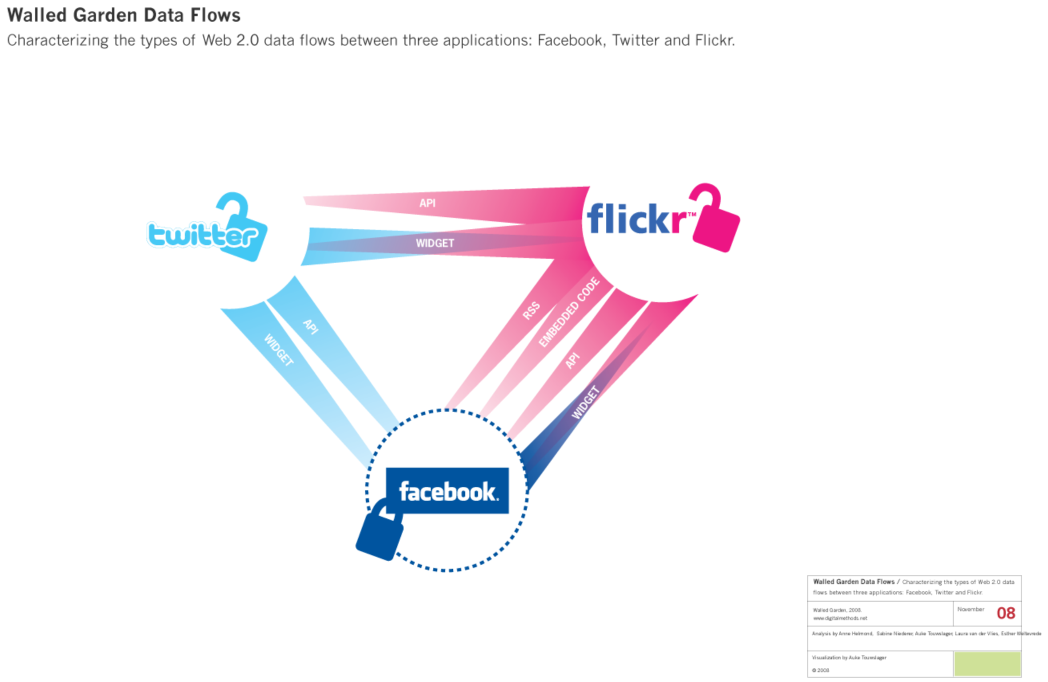

Walled Garden Data Flows
Characterizing the types of Web 2.0 data flows between three applications: Facebook, Twitter and Flickr.
twitter
flickr
facebook
API
WIDGET
API
WIDGET
RSS
EMBEDDED CODE
API
WIDGET
Walled Garden Data Flows / Characterizing the types of Web 2.0 data flows between three applications: Facebook, Twitter and Flickr.
Walled Garden, 2008.
www.digitalmethods.net
November 08
Analysis by Anne Helmond, Sabine Niederer, Auke Touwslager, Laura van der Vlies, Esther Weltevrede
Visualization by Auke Touwslager
© 2008

Figure 3: Walled garden data flows. The graphic characterizes the types of data flows between three social media platforms: Twitter, Flickr and Facebook. It displays the availability of four types of data exchange mechanisms that enable data to flow in and out of the platform: APIs, embed codes, widgets and RSS. The data flows are visualized as inbound and outbound data flows to indicate their direction. The availability of data exchange mechanisms and the direction of data flows have been used to determine the permeability of the platforms. Analysis by Anne Helmond, Sabine Niederer, Auke Touwslager, Laura van der Vlies and Esther Weltevrede. Visualization by Auke Touwslager, created with Adobe Illustrator. © Digital Methods Initiative. November 2008.

For each service we located the features that enable data exchanges by analyzing the user interface as well as reading the developer documentation. We then mapped out the different data exchange mechanisms and their data flows and distinguished between inbound and outbound data flows. What followed from this small analysis was that the extent to which Facebook, Twitter and Flickr allow data to flow in and out of their networks is quite different. In November 2008, we characterized Flickr as the most open service, followed by Twitter, while Facebook was the least open network of the three. As will be argued in chapter 2 and 5, Facebook has since developed into the opposite direction, into a platform based on carefully regulated openness. Analyzing data exchange mechanisms and different types of data flows over time can thus serve as entry points to investigate the changing "politics of platforms" (Gillespie 2010), as I discuss throughout in chapters 2, 3, 4, 5 and 6.

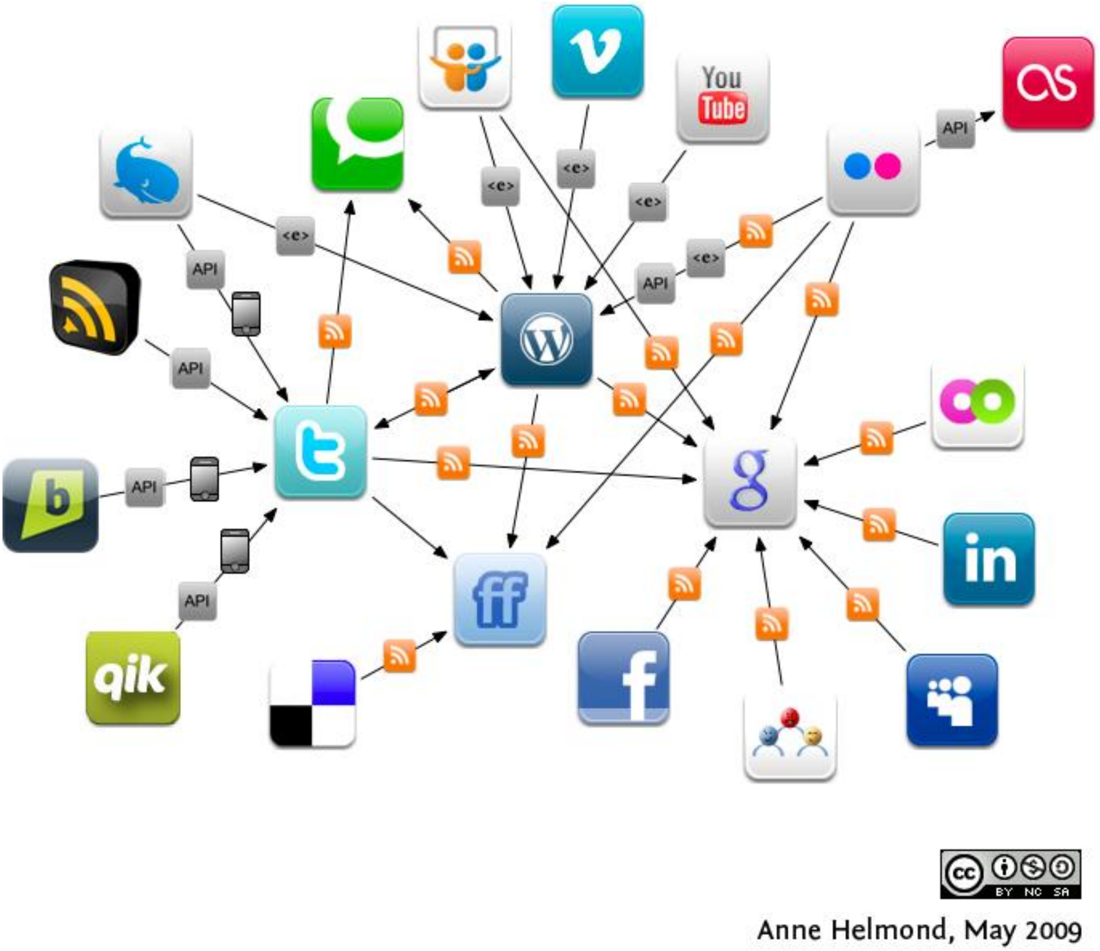


Figure 4: Data flow diagram depicting the various data exchange mechanisms between social media platforms. The diagram represents the data flows between the social media platforms used by a single user, myself, in the Spring of 2009. The graphic shows the interoperability mechanisms—the data exchange mechanisms of APIs, embed codes and RSS feeds—connecting the various platforms into a social media platform ecology. Visualizations by Anne Helmond, created with OmniGraffle. May 2009.

The visualizations made during the Walled Garden conference (see figure 2 and 3) were later expanded in May 2009 to show the interoperability mechanisms between different social media services illustrating how they form social media platform ecologies (see figure 4). Figure 4 resembles what in systems analysis and software engineering is referred to as a "data flow diagram," or a "diagrammatic representation widely used in the functional analysis of information systems" (Batini, Nardelli, and Tamassia 1986). According to Bruza and Van der Weide a data flow diagram (DFD) "can be viewed as directed graph wherein the nodes are external entities, processes or data stores and the edges are data flows" (1993, 67). In his seminal book on *Structured Analysis and System Specification* Tom DeMarco explicates that a data flow diagram is a "network representation of a system" (1978, 47) that focuses on the flow of data through a system and the processes that transform data. This, DeMarco continues, distinguishes data flow diagrams from flowcharts since "[t]he Data Flow Diagram portrays a situation from the point of view of the data, while a flowchart portrays it from the view of those who act on the data" (1978, 40).

While originally conceived as an approach to analyze and design information systems, I have taken up the insight to work "from the view point of the data" (DeMarco 1978, 40) to analyze the role of the platform infrastructure of social media. With its focus on data flows, processes and transformations, the data flow diagram approach may be repurposed as analytical tool for analyzing social media platforms. Here, I would like to acknowledge the work of Ganaele Langlois and her co-authors who aim to understand social media platforms in a similar manner by mapping the connections between users, content and protocols as a form of critical intervention (2009). In this dissertation I contribute to their call for new tools to track, map and visualize the information channels of Web 2.0 to analyze how objects flow through these channels and how they are reconfigured to understand the "technocultural logics" of platforms (2009). This approach is situated within the related fields of software studies and platform studies, as discussed in more detail shortly.

The sketches of data flows diagrams serve as an analytical means to empirically study the role of the infrastructure of social media platforms by focusing on interconnections between services and websites and the data that flows between them. This is shown in chapter 2 by analyzing the components of data exchange mechanisms in the social web, in chapter 3 by mapping out the interconnections and data flows between blogs and social media platforms, in chapter 4 by showing the data flows between the Twitter platform, link sharing services built on top of Twitter and their users, in chapter 5 by tracing the data flows between Facebook, users and external websites that have implemented the Facebook Like button and in chapter 6 by analyzing how we can detect and make use of these third-party connections and data flows to study website ecologies over time.[31] Thus the dissertation introduces a particular mapping practice for data exchange with third parties, flow between blogs and social media, linkage and sharing services, spread of social buttons and ultimately cross-platform ecology.

These visualizations of the interconnections between social media platforms show that they carefully regulate their permeability through a number of different devices that allow data to flow in and out of these platforms to and from other platforms, websites and apps. These mechanisms to connect to other platforms as well as external websites and apps challenge the conceptualization and critique of social network sites as walled gardens, as argued in more

[31] In this dissertation I stop following the data flows when they have reached third parties and do not trace and map how these third-parties may invoke further connections, often referred to as fourth-party connections (Chaabane et al. 2014).

detail in chapter 2. Instead, social media platforms are forming platform ecologies or what in business literature is referred to as the "social media ecosystem" (Hanna, Rohm, and Crittenden 2011) or what José van Dijck calls "an ecosystem of connective media" (2013c, 4).[32] Similarly, I argue for the study of web ecologies in terms of platformization. The ontological distinctiveness of platforms, as I argue in chapter 2, lies in the way they enact their programmability, or the ways in which they make themselves accessible to third parties. This draws attention to the platform-specific objects that enable connections with third parties and thereby entangle platforms, users, webmasters, app developers and other web actors in complex relations.

Resulting from the two related observations on the interconnectedness of platforms and the decentralization of platform features, this thesis thus addresses the following research question: How have social media platforms extended themselves into the web and, what are the consequences of these extensions?

In introducing the topic of the dissertation, I have discussed the urgency of platform studies for the web and its users, but have said little about the scholarship in which I would like to situate the work. In the following, I discuss a number of calls that have been made to study social media platforms, and previously software more generally, so that I may distinguish my own approach within a broader, recent scholarly discourse.

Studying social media platforms: Software studies meets platform studies

In this dissertation I contribute to discussions that call for studying social media platforms in a technical sense,[33] that is, to pay attention to their underlying infrastructure in order to understand how the web has been transformed by social media. These calls come from what I see as two interrelated emerging fields of study: software studies and platform studies. From software studies, I address how one is invited to study not only cultural practices, the shaping of participation, sociality, and effects on users, but also the consequences of the software infrastructure. From platform studies, I address the need to analyze not only how platforms enable and constrain particular use and development practices but also how they employ their software infrastructures to weave themselves into the web and to format external web data according to the logic of the platform.

The first overall call comes from the emerging field of software studies to put software, as an understudied object of study and "a blind spot in the theorization and study of computational and networked digital media" (Fuller 2006), on the agenda of media studies.

[32] See chapter 6 on the use of ecological metaphors in new media studies.

[33] In her dissertation Taina Bucher responds to a number of similar calls "made by media scholars in recent years for a better understanding of software in studies of social media" (2012b, 29). What distinguishes my approach from Bucher is the focus on platforms as a specific type of software, which, I argue, come with their own set of questions and approaches.

This field has emerged from a number of authors who engage with the relationship between software and culture (e.g. Manovich 2001; Fuller 2003; Kirschenbaum 2003; Mackenzie 2006; Fuller 2008; Berry 2011; Kitchin and Dodge 2011; Chun 2011; Manovich 2013).[34] One of the key concerns these authors share is to make software, which "has come to intervene in nearly all aspects of everyday life and has begun to sink into its taken-for-granted background" (Thrift 2005, 153) and to make the work that it does in our world visible.

Recently, various authors have brought a software studies perspective to social media analyzing the role of software within Web 2.0 spaces.[35] Langlois et al. have analyzed how software and cultural practices mutually shape each other (2009, 416; 2009) and how the software and protocols of Web 2.0 platforms establish the "technocultural conditions" for the circulation of content and participation of users (2009). This participation, Mirko Schäfer similarly argues, is channeled through software design with features for ranking, rating and sharing (2011, 51–52). Software, Taina Bucher (2012b) and José Van Dijck (2013a) contend, does not only structure participation but also programs particular forms of sociality.[36] This demonstrates, Van Dijck and Robert Gehl suggest, how software architectures encode cultural norms and values (2013c, 14; 2014). What this dissertation adds to this research on the role of software in the social web is an analysis of the impact of social media platforms on the web's infrastructure.

Drawing from Actor-Network-Theory (ANT) (Latour 2005), [37] authors such as Langlois et al. (2009, 416; 2013, 5), Niederer and Van Dijck (2010, 1373), Schäfer (2011, 17), Bucher (2012b, 193) and Gehl (2014, 12–13) highlight the importance of studying how Web 2.0 spaces, content and activities are co-constituted by human and non-human actors such as software. For that reason, Niederer and Van Dijck propose to examine "the sociotechnical system that lies at the core of Web 2.0 platforms" in order to understand how content and practices are mutually shaped by software and users (2010, 1384). A related call has previously been made by David Beer to "understand how the material infrastructures of Web 2.0 play out in the lives of individual users, how the software constrains and enables, how it formulates hierarchies, shapes the things people encounter, and so on" (2009, 1000). Beer also offers a framework that consists of three analytical levels to begin such inquiries into Web 2.0 infrastructures: The first level concerns the organizational and economic aspects of Web 2.0 applications, the second level their software infrastructures and the third level combines the previous two to analyze their effects on users (2009, 998). Such an approach to social media platforms encourages us to take the role of their infrastructures seriously. However, instead of

[34] For an overview of scholarly research produced under the umbrella of software studies, see David Berry (2011, 4–5).

[35] This is not meant as a complete overview, rather I wish to highlight authors that align themselves with software studies in studying social media, for more see Taina Bucher (2012b, 29–30).

[36] Whilst these perspectives analyze the role of software in creating connections, Tero Karppi focuses on the politics of disconnections within social media to analyze the logic of connectivity (2014).

[37] See Langlois (2005) for an introduction to actor-network theory in relation to medium theory and new media studies.

only concentrating on the effects on web users, I propose to also examine the consequences for the web's infrastructure.

This dissertation takes up the calls made by Beer (2009) and Niederer and Van Dijck (2010) to focus on the material foundations of Web 2.0 to understand the changing nature of the web with the rise of social media. In order to study the substrates of social media, I argue, we need to turn to platforms, as software platforms are the "invisible engines" (Evans, Hagiu, and Schmalensee 2006, vii) that power social media.

In doing so, I move from software studies to the related field of platform studies, as introduced by Ian Bogost and Nick Montfort, in which they position the platform as an understudied level of new media (2007, 1). Bogost and Montfort call for "platform studies" to pay close attention to the underlying computing infrastructures of new media objects, platforms, and to connect their "technical details to culture" (2009). This can be seen as a response to the so-called "vapor theory" in media studies (Lovink 2003, 243) in which new media objects are described in intangible concepts and metaphors.[38] This abstraction, Eugene Thacker warns us, omits "a specific consideration of the material substrate and infrastructure" of new media (in Galloway 2006, xiii). Platform studies is seen as a materialist or media archeological approach to new media objects (Apperley and Jayemane 2012, 10–11) that urges us to move away from abstract ideas about platforms, and instead, take their technical infrastructure into account (Bogost and Montfort 2009).[39]

Thomas Apperley and Darshana Jayemane note that this approach shares a similar concern with software studies, that is, to take the "stuff" that constitutes new media seriously (2012, 11). Since both share a materialist tradition, one would imagine that platform studies would be subsumed under software studies. However, Apperley and Jayemane contend, it is more complicated because Bogost and Montfort seem to differentiate platform studies from software studies "by arguing for a strong separation of code and platform" (2012, 11). Whilst Bogost and Montfort see platform studies as "highly compatible and consistent" with software studies they position the platform as a separate and neglected layer of new media objects (2009). They distinguish between five levels "that characterize how the analysis of digital media has been focused—each of which, by itself, connects to contexts of culture in important ways" (2009, 145). These five levels, and their respective types of analyses are, from top to bottom: reception/operation (e.g. studies of media effects), interface (e.g. interface studies and human computer interaction), form/function (e.g. ludology/narratology), code (e.g. code and software studies) and platform (e.g. platform studies) (2009, 145–147).

[38] See Marianne van den Boomen on the role of metaphors in new media (2014).

[39] This expanding interest in the materiality of the medium has been framed the 'material turn' in media studies. See Jonathan Sterne (2014) for a history and overview of materialist analysis of media technologies and Taina Bucher for an overview on 'Medium theory and the materiality of media' and its relevance for studying social media from a materialist perspective (2012b, 30–35). In 'Game Studies' Material Turn' Apperley and Jayemane describe platform studies' materialist foundation and its connection to media archeology in more detail (2012, 10–11).

In this dissertation I align myself with Apperley and Jayemane who question the strong separation between these levels and who position platform studies as part of software studies (2012, 11). They see platform studies' focus on connecting the technical details of platforms to cultural aspects as part of Lev Manovich's proposal for software studies to study the relations between the "computer layer" and the "cultural layer" of new media (Manovich 2001, 63–65; Apperley and Jayemane 2012, 11). In addition, software platforms such as social media platforms further complicate the alleged distinction between software and platform as, Anastasia Salter and John Murray claim, "[d]emarcating the boundaries of the platform is difficult where software is concerned" (2014, 11).

So far, Salter and Murray argue, the main body of work that has been produced under the umbrella of platform studies—including the Platform Studies book series with MIT Press—has focused on hardware platforms (2014, 9) and in particular video game platforms (Leorke 2012). In his critical account of the Platform Studies series Dale Leorke argues that, in the first book of the series, Nick Montfort and Ian Bogost have introduced a methodological recipe for platform studies, which subsequently has been applied to other video game consoles in the series (2012). This, Dale Leorke claims, "reduced platform studies to a generic formula that can be emulated for any platform that's called for" and may have instilled the idea that platform studies is solely about hardware and video games (2012, 266). In their FAQ about platform studies, Bogost and Montfort have previously addressed these common misunderstandings to emphasize that platform studies "extends to all computing platforms on which interesting creative work has been done," including software platforms (2009). In the latest book in the Platform Studies series, Salter and Murray (2014) make an important contribution to platform studies by moving beyond hardware with their focus on the software platform Flash and by drawing from the related field of software studies. This dissertation further connects the fields of software studies and platform studies by putting software platforms, and in particular social media platforms, on the agenda of platform studies.

Another important contribution towards studying software platforms comes from a number of authors who engage with a platform politics perspective (e.g. Gillespie 2010; McKelvey 2011; Bucher 2012b; Hands 2013; Langlois and Elmer 2013; Van Dijck 2013c; Gerlitz and Helmond 2013; Puschmann and Burgess 2013). Platform politics approaches include critically interrogating the platform concept (Gillespie 2010; McKelvey 2011), analyzing the "technocultural logics" of platforms (Langlois, Elmer, et al. 2009; Langlois, McKelvey, et al. 2009), and examining the role of the platform architecture in shaping networked sociality (Bucher 2012b; Van Dijck 2013c) (see Renzi 2011). [40] In the opening words of the Platform Politics conference, co-organizer Jussi Parikka positioned the idea of

[40] 'Platform Politics' is the title of a conference held at Anglia Ruskin University in Cambridge, UK on May 12-13, 2011. The conference was organized by Josh Hands and Jussi Parikka and brought together a number of scholars studying the politics of platforms. At this conference Carolin Gerlitz and I presented the Like Economy, see chapter 5. Following the conference *Culture Machine* published a special issue entitled 'Platform Politics' (2013), edited by Joss Hands, Greg Elmer and Ganaele Langlois which contains more recent work on platform politics.

platform politics as bringing together Bogost and Montfort's layering model of platform studies: "[…] in addition to the specific level of 'platforms' we can think of the platform itself as distributed on a variety of layers as assemblages" (Parikka 2011). That is, the politics of platforms run through the different levels of new media objects and connects them.

In this dissertation I engage with both Bogost and Montfort's more narrow and computational understanding of platforms as well as the idea of the politics of platforms by critically engaging with the computational infrastructure of platforms to analyze the work that these platforms do (cf. Gillespie 2010). In doing so, this dissertation contributes to both software studies and platform studies by focusing on the effects of the computational infrastructures of social media platforms on the web and its users.

So far I have addressed a number of calls that have been made to take the level of the software platform infrastructure seriously. I now move on to introduce the five case studies in which two related calls are taken into account. These calls pertain to the need for new methodologies in software studies as well as platform studies, including Manovich's call in his book *Software Takes Command* (2013, 15) and Bogost and Montfort's suggestion that the platform layer requires "a willingness to use new and challenging methods of thinking and investigation" by bringing "nuanced cultural analysis to bear on computer systems" to platform studies (2009).

Studying platformization: Methodological considerations

This dissertation is organized around five case studies—discussed in more detail in the chapter overview—in which I analyze the platformization of the web chronologically. These cases have been selected because they illustrate distinct aspects and consequences of the ontological distinctiveness of social media platforms, that is, their programmability as enacted through APIs. Each case is organized around web-native infrastructural elements and emerging platform features in relation to specific web periods. I focus on distinct medium-specific features of social media platforms, or platform-specific objects such as data exchange mechanisms (chapter 2), widgets (chapter 3), shortened URLs (chapter 4), social buttons (chapter 5) and trackers (chapter 6). Each case traces the emergence of these specific platform features and explores the consequences of their introduction for the web's infrastructure as well as users. Social media platforms Facebook and Twitter play a central role in this dissertation to explore these objects because they are the two most visited social media platforms,[41] and because a large number of websites have been integrated with Facebook's and

[41] According to Alexa, Facebook ranks #2 and Twitter #8 in the Top 500 sites on the web, see http://www.alexa.com/topsites [Accessed 18 April 2015]. These rankings are derived from the "average daily visitors and pageviews over the past month" (Alexa).

Twitter's platform.[42] Together, these cases provide a socio-material perspective on social media platforms so as to explore how they are turning other online web spaces into instantiations of social media. Now that I have discussed the selection of the five case studies I move on to outline my methodological approach to my research.

This dissertation aligns itself with software studies, platform studies and digital methods and has learned from their approaches in order to derive a methodology to study platformization. The authors that have been addressed in the previous section employ methods from various disciplines or have derived their own methods to analyze social media since software studies itself does not offer any particular methods. According to Lev Manovich, Matthew Fuller suggests in the introduction of the *Software Studies* (2008) lexicon that software "can be studied using already existing methods—for instance, actor-network theory, social semiotics, or media archaeology" (Manovich 2013, 15). Similarly, Rob Kitchin and Martin Dodge see "a wide range of social science methodologies" that could be used to analyze software (2011, 255–258).

Despite the fact that these existing methods have proved to be very valuable, Manovich contends that software studies is also in need of *new* methodologies (2013, 15) in order to contribute to the development of the field through technical engagements with software. I put forward that technical insights into software platforms, or a deep understanding of the technical workings of software—what Bogost and Montfort refer to as "being technically rigorous" (2009)—can be gained from reading a platform's developer documentation, developer blog, company blog, privacy policy, terms of service or help documentation. In this dissertation I have made extensive use of these source materials to study software platforms and their features. In doing so, I follow Matthew Kirschenbaum's proposal to view and study software as "the product of material environments" (2003), which is worth quoting at length here:

> Software is the product of white papers, engineering specs, marketing reports, conversations and collaborations, intuitive insights and professionalized expertise, venture capital (in other words, money), late nights (in other words, labor), Mountain Dew, and espresso. These are material circumstances that leave material traces - in corporate archives, in email folders, on whiteboards and legal pads, in countless iterations of alpha versions and beta versions and patches and upgrades, in focus groups and user communities, in expense accounts, in licensing agreements, in stock options and IPOs, in carpal tunnel surgeries, and in the [former] Bay Area real estate market (to name just a few) (Kirschenbaum 2003).

Kirschenbaum sees these traces of the production process of software as valuable sources for software studies scholars. They are central to his vision of how critical inquiries into software could be operationalized: "Software studies is, or can be, the work of fashioning

[42] According to BuildWith, 28,6% of the Quantcast Top 10k websites are integrated with Facebook and 14,8% with Twitter, see http://trends.builtwith.com/javascript/Facebook-for-Websites and http://trends.builtwith.com/javascript/Twitter-Platform [Accessed 18 April 2015].

documentary methods for recognizing and recovering digital histories, and the cultivation of the critical discipline to parse those histories against the material matrix of the present" (2003).[43] The material traces of software, web historian Megan Ankerson adds, are important resources for studying changes in and writing histories of the web from a software studies perspective (2009).

A notable example of using material traces to study digital media infrastructures is Alexander Galloway's *Protocol* (2006) in which he analyzes internet protocols as part of a new apparatus of control by examining the documents—the Request for Comments (RFCs)—that detail these protocols (2006). These RFCs contain technical and organizational information about the internet (IETF 2015) and are, Galloway argues, "a discursive treasure trove for the critical theorist" (2006, 38) to examine how "control takes shape in the materiality of networks" (2006, xvi). In this dissertation, social media platforms' own platform documentation has proven to be a valuable source. In addition, I have also relied on external developer blogs, technology blogs, Q&A websites for programmers such as Stack Overflow[44] and the trade press writing about social media platforms and specific platform features, protocols, formats and standards (see footnote 9). To analyze how social media platforms features have developed over time I have also made use of the largest publicly accessible web archive, the Internet Archive's Wayback Machine,[45] to uncover historical platform documentation (see chapter 3).

These material traces of software platforms provide an important entry point for understanding the architecture, features and underlying logics of social media platforms. In this dissertation I compliment this software studies approach of adapting documentary methods (Kirschenbaum 2003) with so-called "platform-specific methodologies" (Langlois, Elmer, et al. 2009) to study social media. Studying social media platforms is a challenging undertaking, Langlois and Elmer argue, because "their logic goes against critical approaches" as these platforms tend to obfuscate their inner workings (2013, 8). In their previous work on social media platforms, Langlois et al. have argued for taking the specificities of platforms into account in order to understand the "technocultural logics" of social media (Langlois, Elmer, et al. 2009, 429; Langlois, McKelvey, et al. 2009; Elmer and Langlois 2013). I take their approaches further by developing a methodological framework to understand how social media has transformed the web by looking at the role of the platform infrastructure. An important methodological aspect in the work of Langlois et al. that I engage with is their observation that with the so-called shift from Web 1.0 to Web 2.0 the connectivity elements

[43] Kirschenbaum has further developed these ideas in his book *Mechanisms* in which he analyzes and reconstructs new media objects through their material traces, that is their "forensic materiality," by applying methods from computer forensics (2007, 20). In doing so, Kirschenbaum argues, "[t]he book's 'forensics' is therefore a theoretically and technically rigorous account of electronic texts as artifacts—mechanisms—subject to material and historical forms of understanding" (2007, 17).

[44] "Stack Overflow is a question and answer site for professional and enthusiast programmers." See: https://stackoverflow.com/ [Accessed 29 July 2014].

[45] See: http://archive.org/web/ [Accessed 1 March 2015]

of the web have changed, requiring us to rethink how we study the web (2009). The web no longer only consists of HTML pages that are connected by hyperlinks but the web is now also connected through software elements (Langlois, McKelvey, et al. 2009) such as social buttons (see chapter 3). For that reason, Langlois et al. argue, we need new platform-specific methods that are attuned to these new Web 2.0 objects and connection mechanisms (2009).

In this dissertation I put forward novel platform-specific methods to trace the infrastructural elements of social media platforms in order to explore how they extend into the web and format external web data. So in order to study platformization, I argue that one should focus on the platform-specific features that enable the programmability of platforms.

I approach this undertaking from a medium-specific perspective by paying close attention to the technical specificity of social media platforms as put forward by Langlois et al. (2009) and Rogers (2013). By employing such a perspective, this dissertation also aligns itself with digital methods (Rogers 2013) as a digital research method to analyze social media platforms. Digital methods approaches focus on natively digital objects that are specific to social media platforms and the methods that are embedded in those platforms (Rogers 2013, 1). In *Digital Methods* (2013), Richard Rogers[46] distinguishes between the natively digital and the digitized in regard to both web objects as well as methods. Natively digital objects are web objects that are native in a computing sense, that is they are 'written' for the medium (Rogers 2013, 19) such as the hyperlink, the permalink, the shortened URL, the social button, the tag and the API. In contrast, digitized objects such as scanned documents have not been created for the web specifically and have 'migrated' to it by scanning and uploading (Rogers 2013, 16–19). Rogers further develops this distinction between the natively digital and the digitized in relation to internet research methods (2013, 19). Digital methods positions itself as an assemblage of research methods that are sensitive to the methods embedded in the medium, in contrast to applying existing methods from the social sciences and humanities (2013, 19). An important aspect of digital methods is to repurpose the methods of the medium and to employ their output for social research questions (Rogers 2013, 15). A leading starting point in this type of research is to examine how web devices such as social media platforms handle natively digital objects and how they can be repurposed for analytical means (Rogers 2013, 26). How can we make use of tweets, retweets, likes and shares for social and cultural questions? My contribution to digital methods has been to employ digital methods techniques not necessarily for social research but for medium research, that is, to study the role of the platform infrastructure.

Whereas a large body of digital methods research focuses on repurposing the output of devices—the data of social media platforms such as Facebook and Twitter—to answer social

[46] Richard Rogers is the director of the Digital Methods Initiative (DMI), the New Media & Digital Culture research program at the University of Amsterdam. This research group is dedicated to reworking methods for internet-related research (Rogers 2013, 19). Since 2007, I have been a founding member of DMI, where my contributions have focused on developing methods to study social media.

and cultural research questions, I have focused on the question how medium-specific objects and methods can be put to use to study platform infrastructure. What are the medium-specific features of social media platforms and how can they be employed to explore infrastructural changes on the web? How can we make use of natively digital objects to understand the role of the platform infrastructure in handling and transforming web objects? So in addition to the treatment of existing web-native objects, I wish to emphasize the transformation of web-native objects by social media platforms into platform-specific objects by studying the differences between the input and output of these platforms. Which existing objects have been transformed by social media platforms, which new platform objects have been introduced in their transformation and with what objectives? In thinking through these questions I would like to suggest that a medium-specific view on platforms, or what Langlois et al. call a platform perspective (2009), introduces the notion of *platform-specific objects* in addition to web-native objects. Such a distinction allows me to draw attention to the role of the platform infrastructure in developing novel methods to analyze the platformization of the web.

In this dissertation I analyze how social media platforms handle natively digital objects such as the hyperlink and also explore how they create new platform-specific objects such as APIs, widgets, social buttons, shortened URLs and trackers. I employ these platform-specific objects for analytical purposes to put forward new methods to study social media platforms and in particular the process of platformization. In chapter 2, I employ the approach of disaggregation (Langlois, McKelvey, et al. 2009) by taking the platform apart to examine its specific components. This enables me to critically inquire into the ontological distinctiveness of platforms, that is their programmability through APIs and associated social plugins, to understand how they are weaving themselves into the web. In chapter 3, I propose novel methods to reconstruct and map a historical blogosphere using Internet Archive Wayback Machine data to analyze changing linking patterns in the blogosphere with the rise of social media. This allows me to analyze how social media platforms have introduced new linking practices through widgets and how these widgets have transformed the hyperlink structure of the blogosphere. In chapter 4, I put forward a method to follow shortened URLs to trace how social media platforms have reconfigured the hyperlink. This enables me to explore the role of the platform architecture in creating new link types to make external web data platform ready. In chapter 5, I develop a method to trace and map the spread of social buttons on websites. This allows me to explore how social media platforms are creating data-intensive infrastructures on the web to transform web activities into valuable platform data. Finally, in chapter 6, I suggest a method to map historical website ecologies, that is, to show the larger techno-commercial configurations that websites have been embedded in over time. This enables me to understand the changing structure of the website in a platformized web as well as the web's composition at large from the viewpoint of the website.

By putting forward these novel digital methods for studying social media platforms I aim to contribute to the emerging fields of software studies and platform studies and specifically their calls for methodological development. In addition, I contribute to digital methods

(Rogers 2013) with methods for what I call platform infrastructure studies. Studying platformization can be seen as a medium-specific approach to studying the infrastructural changes on the web as caused by social media platforms.

Chapter overview

Each chapter provides a case study examining a different aspect of the platformization of the web and its consequences which I discuss in more detail below. The first four case studies (chapter 2, 3, 4 and 5) pertain to the double logic of platformization, which is characterized by the decentralization of platform features and the recentralization of platform-ready data. The first pair (chapter 2 and 3) concerns the extension of social media platforms into the web. The second pair of case studies (chapter 4 and 5) examines how social media platforms are formatting external web data according to the underlying logic of their platforms. The final case study (chapter 6) considers how we can analyze the platformization of the web at large through the changing composition of the website.

In chapter 2 on 'The platformization of the web' I trace the rise of the platform model on the web in relation to the shift from social network sites to social media platforms. Social network sites such as Facebook have often been conceptualized and criticized as walled gardens. In this chapter I inquire whether this conceptualization is still the most pertinent analytical framework to explore infrastructural changes on the web with the shift from social network *sites* to social media *platforms*. I explore this shift with a historical perspective on what I refer to as the platformization of the web, or the rise of the platform as the dominant infrastructural and economic model of the social web. Platformization entails the extension of social media platforms into the rest of the web and their drive to make external web data platform ready. I explore this double logic of platformization by examining how websites have historically enabled their programmability through data exchanges with third parties. Through disaggregation (Langlois, McKelvey, et al. 2009), that is by taking the platform apart into smaller parts, I locate and examine the components that enable its programmability. I trace the evolution of these components for the circulation of content outside of a site's boundaries: XML, widgets and APIs. The analysis shows that the politics of data flows in the platformized web have transformed from open standards for interoperability to proprietary APIs that adhere to the logic of making external web data platform ready. The specific technological architecture of platforms requires a different type of discourse and critique that moves beyond the walled garden metaphor by taking the medium-specificity of platforms, their programmability, into account.

In chapter 3 on 'The coming of the platforms: Rethinking the history of the Dutch blogosphere' I analyze the rise of social media in the Dutch blogosphere and the symbiotic relationship between social media platforms and blogs. The blogosphere has played an instrumental role in the transition and the evolution of linking technologies and practices. This chapter maps historical changes in the Dutch blogosphere and the interconnections between blogs, which—traditionally considered—turn a set of blogs into a blogosphere. I put forward novel methods to reconstruct a historical blogosphere and to analyze changes in the underlying infrastructure of the blogosphere over time. Specific attention is paid to the changing patterns of hyperlinking and especially how social media platforms have introduced different forms of linking. This research aims to repurpose the Internet Archive Wayback Machine to 1) map the rise and fall of the Dutch blogosphere, 2) analyze infrastructural changes in the blogosphere 3) trace and map changing linking technologies and practices in the blogosphere in relation to the rise of social media, and 4) study the role of social media platforms within the Dutch blogosphere. These aspects together have enabled me to map the changing ecology of the blogosphere over time. In this chapter I argue that the rise of social media platforms has fundamentally changed the link structure of the blogosphere and that current hyperlink network analysis tools are not attuned to analyze the platformization of the blogosphere. Therefore, this chapter makes a methodological contribution to hyperlink analysis to map the platformization of the blogosphere.

In chapter 4 on 'The algorithmization of the hyperlink: Making data platform ready' I examine the changing role of the hyperlink with the introduction of social media platforms. In this chapter I have looked into the history of the hyperlink from a medium-specific perspective by analyzing the technical reconfiguration of the hyperlink by web devices such as search engines and social media platforms over time. Hyperlinks may be seen as having different roles belonging to specific periods, including the role of the hyperlink as a unit of navigation, a relationship marker, a reputation indicator and a currency of the web. The question here is how web devices have contributed to constituting these roles and how social media platforms have advanced the hyperlink from a navigational device into a data-rich analytical device. By following how hyperlinks have been handled by search engines and social media platforms, and in their turn have adapted to this treatment, this study traces the emergence of new link types and related linking practices. The focus is on the relations between hyperlinks, users, engines and platforms as mediated through software and in particular on the process of link shortening by social media platforms. The important role these platforms play in the automation of hyperlinks through platform features and in the reconfiguration of the link as database call is illustrated in a case study on link sharing on Twitter. The automated reconfiguration of the link into an analytical device so as to become part of an algorithmic system is what I refer to as the algorithmization of the hyperlink. I demonstrate the changing role of the hyperlink in the web as platform, where the link becomes a database call and a device to make data platform ready. In this chapter I have developed a novel method to follow shortened URLs by mapping their redirect paths to map and analyze the role of the platform infrastructure in turning the hyperlink into an analytical tool.

In chapter 5 on 'The Like economy: Social buttons and the data-intensive web' I examine Facebook's ambition to extend into the entire web by focusing on social buttons and developing a medium-specific platform critique. The chapter contextualizes the rise of buttons and counters as metrics for user engagement and links them to different web economies. Facebook's Like buttons enable multiple data flows between various actors, contributing to a simultaneous de- and recentralization of the web. They allow for the instant transformation of user engagement into numbers on button counters, which can be traded and multiplied but also function as tracking devices. The increasing presence of buttons and associated social plugins on the web creates new forms of connectivity between websites beyond hyperlinks, introducing an alternative fabric of the web. This chapter presents a novel method that repurposes the browser plugin Ghostery to map these data connections which are creating a data-intensive infrastructure on the web. Whereas Facebook claims that social buttons promote a more social experience of the web, this chapter argues that these buttons are part of a technical infrastructure in which social activities are turned into valuable data, conceptualized as a so-called "Like economy". I argue that the platformization of the web has shifted the currency of the web from web-native links to platform-native likes.

Chapter 6 on 'Website ecologies: Redrawing the boundaries of a website' builds on the previous chapters and shows that websites in the web as platform have become increasingly shaped by dynamically-generated third-party objects and functionality such as embedded content, social plugins and advertisements. This draws attention to the larger techno-commercial configurations of the web that these websites are embedded in. In these arrangements data flows in structured exchanges between websites, users and platforms, as well as third parties such as advertisers, tracking companies and social media aggregators. In this chapter I reconceptualize the study of websites as website ecology which analyzes how various relations between various actors on the web have become inscribed in a website's source code. The archived website as an object of study becomes a site to study the changing ecologies of the web. In this chapter I present a methodology to reconstruct historical website ecologies by using the source code from archived websites from the Internet Archive Wayback Machine. That is, I employ third-party objects in websites to study a website's ecosystem as a way to examine the spread of platformization. In addition, I put forward a way to employ the affordances of social media platform APIs to retrieve missing platform content in archived websites.

Platform infrastructure studies

In this dissertation I develop a critique of the shift from social network sites to social media platforms through the notion of platformization and its consequences in five case studies. The dual logic of platformization—the decentralization of platform features and the recentralization of platform ready data—draws attention to the role of the platform infrastructure in social media's distribution across the web. In particular, examining platformization shows how social media platforms have developed platform-native objects to weave themselves into the web to turn other online web spaces into instantiations of social media. This dissertation is a contribution to the emerging fields of software studies and platform studies via digital methods to study the effects of social media on the web's infrastructure. Ultimately, I propose a new branch of platform studies that I call platform infrastructure studies, which analyzes the ecosystem of software platforms with digital methods, to which I will return in the conclusion in more detail.

2. The platformization of the web

Social networks such as Facebook have often been conceptualized and critiqued as so-called "walled gardens" (Arora 2014; Berners-Lee 2010; boyd 2007a; Bruns 2008b; McCown and Nelson 2009; Olsen 2006; Quiggin 2013, 96; Rogers 2013, 159; Van Dijck 2013c, 167). Media and communications scholar Payal Arora describes how the spatial metaphor of the walled garden functions as a way to comprehend the architecture of social networks in terms of accessibility, inclusion and exclusion and commodification (2014). She argues that the metaphor is employed to understand social networks as enclosed online spaces in relation to the privatization of the web (2014, xv). This privatization is often framed as a move away from the open web towards closed web platforms and apps (Anderson & Wolff, 2010; Berners-Lee, 2010; Zittrain, 2009). Tim Berners-Lee, founder of the world wide web, describes the open web as a web built on open web standards in contrast to sites built with proprietary standards which create closed spaces (2010, 83). So the idea of an increasingly closed web partially stems from this critique of social network sites as walled gardens.

In this chapter I inquire whether the conceptualization of social network sites as walled gardens is still the most pertinent analytical framework to explore infrastructural changes on the web with the shift from social network *sites* to social media *platforms*. I explore this shift with a historical perspective on, what I refer to as, platformization, or the rise of the platform as the dominant infrastructural and economic model of the social web and its consequences. Platformization entails the extension of social media platforms into the rest of the web and their drive to make external web data platform ready. The specific technological architecture of platforms, I argue, requires a different type of discourse and critique that moves beyond the walled garden metaphor by taking the medium-specificity of platforms into account. In doing so, I follow Ganaele Langlois et al.'s call for a "platform-based perspective" (2009), which, according to Fenwick McKelvey, should critically inquire into the programmability of platforms (2011). I position platformization as a form of platform critique that inquires into the dynamics of the decentralization of platform features and the recentralization of platform data as a way to examine the consequences of the programmability of platforms.

The chapter is organized as follows: First, I focus on the concept of the platform as the prevalent way to conceptualize social media. However, before the term "platform" became a dominant concept social network sites were often discussed in terms of walled gardens and I start there by situating the walled garden concept within a larger historical trajectory of open and closed information system architectures. Next, I focus on the different aspects of walled garden critiques by focusing on Facebook, the social network most frequently associated with the term walled garden. Interestingly enough, Facebook founder Mark Zuckerberg has always been reluctant to call Facebook a social network and instead positions it as a platform (Arrington 2008; Locke 2007). This statement is seen within the context of Tim O'Reilly's idea of Web 2.0 as the web as platform and the rise of the platform infrastructure on the web. I locate the platformization of the web in the moment that all major social network sites have started to offer Application Programming Interfaces, APIs. APIs make website data and functionality accessible to other services and, I argue, have turned social network *sites* into social media *platforms*. I trace this platformization by examining how websites have historically enabled their programmability through data exchanges with third parties and the politics of these data flows.

I pose that the new architectural model of the platform has challenged the conceptualization of social network sites as walled gardens. It has explicitly opened up websites by enabling their programmability with a software interface for third parties. While this could be seen as the walled garden becoming more porous, I rather draw on Alan Liu's notion of "data pours" to understand platforms as pouring data systems that set up data channels to enable data flows between the platform and third parties. I conclude by describing how these data pours do not only set up channels for data flows between social media platforms and third parties, but also how these data channels make external web data platform ready.

A material-technical perspective on social media platforms

The term "platform" has become the dominant concept for social media companies to position themselves in the market and address users and has been widely taken up by consumers, the trade press and academics (Gillespie 2010, 348). Within new media studies the platform concept has gained prominence to draw attention to the "discursive work" they undertake (Gillespie 2010, 348) and to the role of software—which powers social media—in shaping participation and sociality (Langlois, McKelvey, et al. 2009; Bucher 2012b; Van Dijck 2013c; Hands 2013; Clark et al. 2014).

In one of the most central discussions on platforms, Tarleton Gillespie puts forward a rather open account of platforms by focusing on the different connotations of the term (2010). In the computational sense Gillespie defines a platform as an infrastructure to build applications on (2010, 349). However, Gillespie contends, Web 2.0 companies have introduced a broader meaning of the term platform that moves beyond its computational meaning (2010, 351):

> This more conceptual use of 'platform' leans on all of the term's connotations: computational, something to build upon and innovate from; political, a place from which to speak and be heard; figurative, in that the opportunity is an abstract promise as much as a practical one; and architectural, in that YouTube is designed as an open-armed, egalitarian facilitation of expression, not an elitist gatekeeper with normative and technical restrictions (Gillespie 2010, 352).

Gillespie argues that the more conceptual use of the term enables platforms to bring various actors together. They address developers with the computational meaning and evoke the other connotations to address other actors such as users, advertisers and clients (2010, 352). Gillespie actually describes what in economics Jean-Charles Rochet and Jean Tirole call the business model of a "multi-sided market" in which a platform enables interactions between two or more distinct parties (2003, 990).[47] Facebook is an example of a multi-sided platform that creates value by connecting users, advertisers and third-party developers and is characterized by network effects, as value increases for all parties as more people use it (Hagiu 2014).

In his work Gillespie emphasizes the participatory and economic aspects of platforms over their computational dimension by stating that "'[p]latforms' are 'platforms' not necessarily because they allow code to be written or run, but because they afford an opportunity to communicate, interact or sell" (2010, 351). Other authors, such as Ian Bogost and Nick Montfort, suggest a more narrow focus on platforms by foregrounding their computational aspect (2009). In what follows I am interested in developing such computational account of platforms further to examine the work that platforms do (c.f. Gillespie 2010) from a material-technical perspective.

Bogost and Montfort refute the idea that "everything these days is a platform" and call for taking platforms as computational infrastructures seriously (2009). As addressed in the introduction, they see the platform, in its computational sense, as an understudied layer of new media (2009). To address this blind spot, Bogost and Montfort introduce "platform studies," a call for a "technical rigor and in-depth investigation of how computing technologies work" to analyze "the connection between technical specifics and culture" (Montfort and Bogost 2009, vii). Following Bogost and Montfort's call, I wish to draw attention to the importance of analyzing technological platforms in the computational sense.

Previously, in a special issue on *Platform Politics,* edited by Joss Hands, Greg Elmer and Ganaele Langlois, a number of authors have examined "the technological affordances of platforms in relation to their political, economic and social interests" as an important site where "platform politics" play out (Hands 2013; Langlois and Elmer 2013). In this issue, Taina Bucher examines how Facebook employs its platform infrastructure to collect, process and manage user data for organizing attention within the platform (Bucher 2012a, 7). Likewise, I am interested in the role of platform infrastructure, not in "shaping forms of sociality" (Bucher 2012a, 7), but in the reshaping the web as a consequence of the programmability of platforms.

My approach is based on what Langlois et al. refer to as "disaggregation" as a way to critically examine social media platforms by taking them apart and inquire into their specific components (Langlois, McKelvey, et al. 2009). In this chapter I trace and focus on the

[47] While it is outside of the scope of this dissertation to engage with this perspective, Bernhard Rieder and Guillaume Sire make an important call for extending our current analysis of platforms within new media studies with such insights from microeconomics (2013, 197). Studying platforms as multi-sided markets, they argue, "can extend analyses of concrete configurations of power and identify control points, structural dynamics and crucial resources for argumentation" (Rieder and Sire 2013, 208).

elements that make a platform programmable to analyze the politics of platforms through their programmability. This contribution to platform studies and social media studies lies in a detailed material-technical perspective on the development and emergence of what we understand as social media platforms today. Before the platform concept gained prominence however, social media such as Facebook were often discussed as walled gardens, as social networks.

Walled gardens: On open and closed information technologies

In the late 80s and early 90s, the walled garden concept was introduced to describe the proprietary and pre-web networks of online service providers such as CompuServe, Prodigy and AOL (Anderson, 2012, p. 58; Quiggin, 2013, p. 96; Zittrain, 2009, p. 29). According to communications scholar Patricia Aufderheide, AOL pioneered the walled garden model online as a business strategy which "attempts to turn users of networked communications into customers of a proprietary environment" (2002, 518). AOL adopted this model by providing its own content and services within the confines of its proprietary network (Wu 2011, 262). However, this closed model would be challenged by the increasing popularity of internet services like the world wide web.

Initially, AOL "refused its subscribers access to the Internet beyond its walled garden" (Wu 2011, 265) and it was not until 1995 that it started to offer partial access to the web. AOL presented the web as an unruly space and mediated the web experience of its members by incorporating the web into the structured AOL environment to homogenize the online experience (Patelis 2000, 52–53). AOL's integrated browser did not give full access to the web, but instead, provided restricted access to web content to "enhance" AOL's content channels in order to make "the Web feel like part of AOL, rather than like a foreign land" (AOL Annual Report 1995, 11). So while AOL partially opened the gates of its walled garden to let web content in, it reformatted this external web content to make it part of AOL. Eventually, the rise of broadband services which gave direct access to the internet and the popularity of the web contributed to the decline of AOL as a service provider (Wu 2011; Zittrain 2009, 254).

In the mid 2000s, the walled garden concept gained traction again with the rise of social network sites as "virtual gated communities" (Watkins 2010, 68). Primarily Facebook was understood as a walled garden because it was initially only available to Harvard students (boyd and Ellison 2008; Tufekci 2008).[48] The idea of web technologies such as social network sites

[48] With over 581,000 hits on Google and 2,350 results on Google Scholar on January 15, 2015, Facebook has become the most often associated social network with the term walled garden. Facebook was chosen as the prime example because it is the number one social network according to web analytics provider Alexa and it is most associated with the term walled garden on Google and Google Scholar.

as walled gardens has a longer history in the characterization of computing architectures in terms of open and closed.

The walled garden business strategy is based on a closed architecture to maintain control over a system and its users (Aufderheide 2002) and can be traced back to computer systems and economics literature that address vendor lock-in. Economists Paul David (1985) and Brian Arthur (1989) describe how a technology may become dominant because of economies of scale and network effects causing lock-in. If the costs of switching to a different technology are too high this may even lead to being locked into an inferior technology such as the QWERTY keyboard (David 1985). Within the computer industry the history of vendor lock-in is often traced back to IBM's System/360 mainframe (Moschella 1997, 5). The IBM System/360 was a family of compatible computers for which software could be written once and scale to different machines (Steinmueller 1996, 24). It was an attractive investment because customers could purchase a small machine and easily expand or upgrade because of IBM's compatibility approach. A second issue that contributed to vendor lock-in was IBM's decision to bundle the software of System/360 with the hardware rental as part of IBM's business model (Zittrain 2009, 12). The eventual unbundling of hardware and software opened the system up to third-party developers and gave birth to the software industry (Steinmueller 1996, 25).

In his book *The Master Switch*, law scholar Tim Wu describes how information systems, such as the IBM System/360, are often going through cycles of open and closed phases (2011, 6). He sees this pattern repeating in relation to the open internet which is threatened by forces of centralization and closed networks (Wu 2011). Similarly, law scholar Jonathan Zittrain provides a historical trajectory of what he calls "generativity" to describe the openness of systems (2009, ix). According to Zittrain, the PC and the internet are generative systems because "they were designed to accept any contribution that followed a basic set of rules" and claims that this openness fosters innovation (2009, 3). However, Zittrain sees the generativity of the internet as being undermined with the move towards "appliancized networks" that enact a specific model of lockdown with central points of control over content and access (Zittrain 2009, x & 8). In an interview, Zittrain describes how "[t]he future of the Web may be its past: an abandonment of open standards and services [...] and a return to the gated communities that offered consistency and security–and also lock-in" (as cited in Grifantini 2008).

The growth and success of the web has been attributed to its open architecture which is based on egalitarian principles (Berners-Lee 2010, 80). From the beginning, the technical design of the web has been associated with the idea of openness since it has been built on the egalitarian principles of universality and interoperability (Berners-Lee 2010, 82; Bodle 2011, 324). Universality means that anything can link to anything and this principle has guided the design of open standards and protocols for the web (Berners-Lee 2010, 82), encouraging interoperability on the web (Bodle 2011, 325). Within systems engineering interoperability refers to the principle of making system connections compatible (Geraci 1991, 217). Social media platforms such as Facebook, Bodle contends, employ interoperability to facilitate participation and sharing between platforms and third-parties whilst at the same time using these mechanisms to achieve market dominance and lock-in thereby challenging the openness of the web (2011, 328). The open architecture of the web forms the foundation from which more recent walled gardens critiques are formulated.

An important aspect is that access to social network sites is usually password-protected (McCown and Nelson 2009). Most or all social network data and functionality is available to logged-in users and authenticated apps only as the password is key to entering the walled garden. [49] The password-protected environment of Facebook enables users to "construct a notion of 'private' and, even more importantly, contextualized publics" (boyd 2007a). In addition, users employ privacy and visibility settings to further "wall off" their profiles from other users (Tufekci 2008, 34).

Second, they are characterized by data lock-in since all data and content are locked into the network (McCown and Nelson 2009, 251). Even if APIs exist, they are often tightly managed and enforce restrictions on access to data (Bucher 2013). Tim Berners-Lee describes how the architecture of social networks turns them into "closed silo[s] of content" (2010, 82). If a user uploads a photo to Facebook, the connection between the user and the photo exists within Facebook's platform only. In addition, a password is required to enter the gate of the walled garden to access the photo. While the photo has a link, this link identifies the object in Facebook's database and can only be accessed by a logged-in Facebook user or authenticated app with the right permissions.[50] Tim Berners-Lee argues that the creation of these so-called data silos, which are walled off from the rest of the web, pose a threat to the openness of the web (2010). As users become locked into these sites, Berners-Lee continues, "the more the web becomes fragmented" and less of "a single, universal information space" (2010, 82).

Third, related to the issue of data lock-in is the lack of data portability and interoperability. Tim Berners-Lee sees the development of common protocols to enhance the interoperability of the web as an important way to maintain the web's universality principle (2000, 94). In addition, he argues for keeping "the conduit separate from the content" so the web does not become fragmented and therewith ceases to be universal (2000, 94). However, many social network sites threaten these ideas by disabling interoperability through "incompatible data formats and noninteroperable access protocols – that enclose isolated user communities" (Bruns 2008b). Robert Bodle argues that this is often used to support anti-competitive practices against other platforms, where each platform tries to control their data flows by locking in users or by shutting down access to user information or data flows from

[49] Here, I distinguish between logged-in users who gain access to content and functionality in the front-end, Facebook's interface, and authenticated apps that gain access to content and functionality in the back-end, Facebook's database, through the API. It is important to note that the content and functionality that is available through the front-end and the back-end is different.

[50] A link to a photo in Facebook specifies the photo's ID number in Facebook's database, e.g.: https://www.facebook.com/photo.php?fbid=10152937045890586&set=pcb.10152937056105586&type=1

rivals (2011, 323).[51] Whereas interoperability refers to the compatibility of systems, the related idea of data portability refers to the ability to reuse one's own profile and data across various social network sites (Bojars et al., 2008, 6). While Facebook allows you to download parts of your own data,[52] it cannot be imported into and used in another network. This lack of data portability is seen as another threat to the web's universality as it locks users into a particular platform.

Fourth, social networks often do not allow search engine crawlers to index them (Stross 2009, 30; Tufekci 2008, 23). Facebook is largely inaccessible to search engines like Google because it prohibits search engine crawlers from indexing its content.[53] Billions of pieces of proprietary data are shielded off from search engine crawlers by what *Wired Magazine* refers to as 'The Great Wall of Facebook' (Vogelstein 2009). This wall is created by Facebook's robots.txt, a text file on Facebook's server that contains instructions for search engine crawlers about which parts of the site may or may not be indexed.[54] In a series of articles on 'The Battle for the Internet' by *The Guardian*, Google co-founder Sergey Brin portrays the walled gardens of Facebook and Apple as "a threat to the freedom of the web" undermining the internet's principles of openness and universal access (as cited in Katz 2012). Of course, Brin's claims should be seen in the light of the fact that Facebook challenges Google's indexing practices which are based on openly crawlable content.[55]

These critiques of Facebook as a walled garden address the architecture of the social network in its early stages, however when we look at Facebook today, they only still party apply. Whereas Facebook was initially designed as a closed system and has operated as such for years, it has developed itself into the opposite direction by complementing its data centralization with carefully regulated openness (cf. Bodle 2011). This strategic opening, I argue, is the result of Facebook's development as a platform.

[51] For example, six months after Facebook acquired Instagram, competitor Twitter disabled Instagram's photo integration. This is why authors like Anja Bechmann favor the term "intraoperability" over "interoperability" in relation to social media platforms to indicate that these platforms are not involved in symmetrical power relations (2013, 75).

[52] https://www.facebook.com/help/405183566203254 [Accessed 17 March 2015].

[53] Search engines crawlers are permitted to index public Facebook content which includes basic profile information and posts on Facebook Pages and public groups, see: https://www.facebook.com/help/203805466323736 and https://www.facebook.com/help/186212491428940 [Accessed 17 March 2015].

[54] Robots.txt is an advisory protocol but honored by all major search engines.

[55] In the context of Facebook it is currently unknown what percentage of content shared on Facebook is posted as public and can be crawled.

Facebook: Social network site or platform?

Facebook has often been treated as a social network site, defined by boyd & Ellison as:

> web-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).

boyd & Ellison prefer the term "social network site" over "social networking sites" because the former emphasizes how these sites primarily articulate existing social networks while the latter focuses on the act of networking, on finding new connections (2008, 211). In the early days of Facebook, the site focused on existing connections by design as group membership was bound to a user's university address and full user profiles were only available to members of the same college network (boyd and Ellison 2008, 218; Tufekci 2008, 22).

However, Facebook has always carefully refrained from calling itself a social network (Arrington 2008; Locke 2007). Rather, over time Facebook founder Mark Zuckerberg has framed Facebook as a "social directory" (Facebook Newsroom 2006), a "social utility" (Facebook Newsroom 2006) and a "platform" (Facebook Newsroom 2007). In his book *The Facebook Effect* on the history of Facebook, author David Kirkpatrick describes how Zuckerberg has always envisioned Facebook as a "platform" for other applications to run on since its inception as TheFacebook in 2004 (2010, 215–217):

> He [Zuckerberg] wanted to do for the Web what Gates did for the personal computer: create a standard software infrastructure that made it easier to build applications—this time, applications that had a social component. 'We want to make Facebook into something of an operating system, so you can run full applications,' he [Zuckerberg] explained (D. Kirkpatrick 2010, 217).

In the Fall of 2004, Zuckerberg was working on another software project alongside TheFacebook called Wirehog, "a peer-to-peer content-sharing service" (D. Kirkpatrick 2010, 44). Wirehog was integrated into Thefacebook to make use of its friendship connections to share content in TheFacebook with friends. Zuckerberg saw it as "the first example of treating Thefacebook as a platform for other types of applications" (Kirkpatrick 2010, 99–100). So instead of a social network, Mark Zuckerberg has seen and has come to realize Facebook as a platform from the beginning. Facebook's development as a platform should be perceived in the wider context of Web 2.0 as "the web as platform" (O'Reilly 2005), in which the web was positioned as development platform.

Web 2.0: The web as platform

Social network sites are seen as a specific type of Web 2.0 application (Beer and Burrows 2007) or type of social media (Van Dijck 2013c, 8). Web 2.0 or "the participatory web" is often understood as a wide set of applications that foster collaboration and participation (Madden and Fox 2006). The term was popularized at the first Web 2.0 conference in 2004, when Tim O'Reilly rhetorically repositioned the web as "Web 2.0." In the opening lecture, he defined Web 2.0 as "the web as platform," a phrase used to situate the web as a "robust development platform" in which "websites become software components" (O'Reilly and Battelle 2004).

O'Reilly puts the computational meaning of the term platform at the center of the web as platform concept. With Web 2.0, the web as platform, O'Reilly no longer saw the web just as a medium for publishing information—which he retrospectively labeled Web 1.0—but as an infrastructure to build applications on, an operating system that could deliver software services (2005). Therefor, Matthew Allen argues, we should see Web 2.0 as "rhetorical technology" in which "the computing industry attempted to change the way we think of the internet" (2013, 264). This new narrative of the web positioned the web as a software development platform and not just as a new publishing channel (Allen 2013, 264).

However, this original and more computational definition of Web 2.0 as "the web as platform" did not catch on after the conference, Robert Gehl argues (2010, 26–37). Instead, Gehl claims, Web 2.0 was seen as a revival of the industry after the dotcom crash and, even more so within public and academic debates, as a revolution that would reshape the media landscape (2010, 26–37). Web 2.0 technologies were seen as blurring the boundaries between production and consumption (Bruns 2008a), giving rise to new forms of user participation as part of an online "participatory culture" (Jenkins 2006). So while the original definition of Web 2.0 implied making use of the web as a computational platform, it would be embodied in a more metaphorical sense (c.f. Gillespie 2010), as a platform for participation with the associated rhetoric of "empowerment" and "democratization" (Beer 2009, 986).

In order to shift the focus from this broader conceptual notion of platforms back to a more narrow computational understanding I wish to further explore the technological development of software platforms on the web and in particular social media platforms. This allows me to examine the consequences of Facebook's development not as a social network site but as a social media platform.

From social network sites to social media platforms

In the next part I discuss how websites can become platforms by attending to another computational definition of platform, provided by Netscape founder Marc Andreessen in a blog post discussing Facebook's new platform:

> Definitionally, a 'platform' is a system that can be reprogrammed and therefore customized by outside developers -- users -- and in that way, adapted to countless needs and niches that the platform's original developers could not have possibly contemplated, much less had time to accommodate (Andreessen 2007a).

For Andreessen the key term in this definition of a platform is programmable, which eradicates the more conceptual uses of the term: “If you can program it, then it's a platform. If you can't, then it's not” (Andreessen 2007b).

The programmability of Web 2.0 platforms, so Fenwick McKelvey argues, offers a novel line of criticism within platform studies that starts with asking how a platform enacts its programmability (2011). So how does a social media platform *become* a platform? Here, I draw from Evans et al.’s definition of software platform as “a software program that makes services available to other software programs through Application Programming Interfaces (APIs)” (2006, vii).

In order to become a platform, a software program needs to provide an interface that allows for its (re)programming. This interface is called an Application Programming Interface (API) which provides developers access to data and functionality:

> An API is an interface provided by an application that lets users interact with or respond to data or service requests from another program, other applications, or Web sites. APIs facilitate data exchange between applications, allow the creation of new applications, and form the foundation for the ‘Web as a platform’ concept (Murugesan 2007, 36).

Returning to O’Reilly’s positioning of the web as a development platform for new services, these services themselves can also become platforms by providing an API.[56] One of the prime sources on APIs on the web, the website Programmable Web, explains how APIs are at the center of turning websites into platforms: ‘The phrase ‘web as platform’ refers to fact that as web sites start providing their own APIs, they too are becoming a platform on which other programs can be built” (Programmable Web, n.d.). This has been framed as a shift “[f]rom web page to web platform” since “the nature of what a site can be has changed. Rather than being part of a publishing system, Web sites are becoming programmable, much like a PC's operating system” (LaMonica 2005). For example, photo sharing website Flickr is a platform because it offers an API that can be used by developers to build new applications on top of Flickr. A developer can write a script that requests the latest 100 images tagged with ‘flower’ from the Flickr API and display these images on her own website.

This means that a website can have two different interfaces: a user interface for human consumption and a software interface for machine consumption. This software interface, the API, makes a website programmable by offering structured access to its data and functionality and turns it into a platform that others can build on. To extend this line of thinking further, I place APIs at the core of the shift from social network *sites* to social media *platforms*. The moment social network sites start to offer APIs, I argue, they turn into social media platforms by enacting their programmability.

[56] This draws our attention to the use of platform as a recursive concept, as put forward by software developer Dave Winer, whom from early on saw the internet as a meta-platform or ‘platform machine’ which can be used to build new platforms on (1995).

The platformization of the web

Within the field of media studies social media APIs have been understood as the technological glue of the social web in connecting services and enabling the sharing of content (Langlois, McKelvey, et al. 2009; Bucher 2013; Bodle 2011), as protocological objects (Bucher 2013), as regulatory instruments that govern the relations between the platform and third parties (Puschmann and Burgess 2013), as the business model of the social web[57] (Bodle 2011; Bucher 2013) and as tools that construct data for the data market (Vis 2013). Most prominently, APIs have been used and discussed as "a method for data collection on social media platforms" (Lomborg and Bechmann 2014). Less attention has been paid, however, to the history of social media APIs,[58] that is, their emergence on the web as part of the material infrastructure of social media platforms and the consequences of the adaptation of the platform model. One of the most comprehensive accounts so far has been documented by technology blogger Kin Lane who brands himself as "API Evangelist" and who has been studying "the business and politics of APIs" since 2010.[59]

Lane traces the historical emergence of web APIs that target external developers back to the early 2000s, when Salesforce (1999), eBay (2001) and Amazon (2002) started to offer APIs as business-to-business solutions for e-commerce (2012). This first generation of web APIs, mainly provided by e-commerce companies, focused on exchanging data between different business applications to enable transactions and sales management (Lane 2012). For example, Amazon's Web Services platform enabled third-party websites to search their catalogue, display Amazon products and earn referral fees from purchases from their own sites (Amazon 2002). In doing so, Amazon used their API to extend their service into other websites. In the mid 2000s a new generation of web APIs, provided by social network sites, shifted the focus from sales transactions to access to user generated content, user information and their connections (Lane 2012).

In 2003, social bookmarking site del.icio.us started offering programmatic access to its site, followed by Flickr (2004), YouTube (2005), Last.fm (2006), Facebook (2006) and

[57] As part of what the industry calls the "API Economy" to refer to "[t]he emerging economic effects enabled by companies, governments, non-profits and individuals using APIs to provide direct programmable access to their systems and processes" (Willmott and Balas, n.d.).

[58] A notable exception within media studies is the work of Taina Bucher on 'Objects of Intense Feeling: The Case of the Twitter API' (2013) in which she provides a historical background of the role of APIs in software engineering and briefly discusses early public web APIs.
In addition Yu and Woodard analyzed "the structure and dynamics of the Web 2.0 software ecosystem by analyzing empirical data on web service APIs and mashups" (2009). They traced the evolution of this ecosystem using the API repository provided by the website http://www.programmableweb.com/ [Accessed 16 March 2015]. ProgrammableWeb is one of the prime resources for API statistics and they have maintained an API directory since 2005. The directory is updated daily and currently lists over 13,000 APIs. Their mashup directory lists applications that have been built on top of these APIs. In 2002 ProgrammableWeb published an API timeline, which has been valuable for this research, showing the growth of APIs per year and highlighting the introduction of particular APIs (DuVander 2012).

[59] See http://kinlane.com/about/ and http://history.apievangelist.com/ [Accessed 16 March 2015].

Twitter (2006) after which many other social network sites announced their APIs (Lane 2012; DuVander 2012). Robert Bodle describes how these sites made their content and functionality available as part of a business strategy in which third parties can add value to a platform by building new services on top of it (2011, 325). He explains how Tim O'Reilly advocated businesses to pursue a platform strategy by opening up their valuable data to achieve platform lock-in (2011, 325). In his Web 2.0 manifesto O'Reilly further encouraged the reuse of data with the recommendation to "design for 'hackability' and "remixability'" by offering third parties access to data and services (2005). O'Reilly positioned data as the "building blocks" of Web 2.0 (2005) and the availability of data through APIs has made websites programmable. The access to data through APIs has given rise to the typical Web 2.0 practice of creating mashups, that is building new applications by remixing data and functionality from existing sources using APIs (Benslimane, Dustdar, and Sheth 2008). Web 2.0 has therefore also become known as "the programmable web" (O'Reilly 2005; Anderson 2012) in which websites have become platforms by offering interfaces that offer access to their data that can be reused.

Marc Andreessen describes how the programmability of internet-based software platforms can be facilitated on three different levels, producing what he sees as three types of internet platforms. Andreessen describes that most social media platforms provide a so-called Level 1 or "Access API." Here, external developers can access a platform's data and functionality by making API calls (Andreessen 2007b). The API is accessed "from outside the core system" which means that "the developer's application code lives outside the platform" (Andreessen 2007b). Photo sharing service Flickr is an example of an Access API, where a developer can build a third-party application such as a slideshow viewer to show photos tagged with "sunset" by using the Flickr API to access this data. In this scenario the code of the application is located on an external server and the application outside of Flickr. The programmability of a Level 1 platform is characterized by access to data and functionality to create new applications. This means that developers can build new applications on top of the platform but not reprogram the platform itself.

The second approach, the Level 2 "Plug-In API" allows developers to "build new functions that can be injected, or 'plug in', to the core system and its user interface" (Andreessen 2007b). Andreessen uses Facebook as an example of a Plug-In API since it does not only allow developers to access data and functionality from Facebook to build new applications (similar to an Access API) but it also allows for loading and using this application within the Facebook environment itself.[60] However, the code for this third-party application is still located outside of the Facebook platform (Andreessen 2007b).[61]

In the third platform approach, the Level 3 "Runtime Environment," the third-party applications run within the runtime environment of the platform itself (Andreessen 2007b). Andreessen explains that this approach is most similar to 'traditional' computing platforms with an operating system such as Windows where developers built applications that are executed within Windows itself (Andreessen 2007b). The platform as runtime environment is the least common approach on the web since it requires offering a more complicated environment with a technical framework for developers as well as database and storage management (Andreessen 2007b).[62] The programmability of social media platforms is typically enabled through an Access API or Plug-In API. More specifically, in the terms of Andreessen, the most common type of social media platform is the Level 1 Access API (Twitter, Facebook, Tumblr, Instagram), followed by the Level 2 Plug-in API (Facebook).[63]

By distinguishing between different types of platforms Andreessen demonstrates how individual platforms may be critiqued according to their level of programmability. In what follows I develop a platform critique that revolves around the notion of 'platformization' to study and develop a critique of the programmability of social media platforms through their extension into the web. I start by tracing how websites historically have enabled their programmability through the exchange of data, content and functionality with third parties thereby outlining three pre-conditions for the platformization of the web.

The separation of content and presentation

Most websites are created using the HyperText Markup Language (HTML), which describes the content and presentation of a web document. Since HTML is a presentation technology designed for human consumption and many HTML websites are ill-formatted, it is difficult for a machine to extract and process structured information from a website (Myllymaki 2002,

[60] In their developer documentation Facebook explains how this works: "Apps on Facebook are web pages loaded into a Canvas frame. The Canvas frame is simply a blank canvas within Facebook on which to run your app. You populate the Canvas frame by providing a Canvas URL that contains the HTML, JavaScript and CSS that make up your app. When a user loads your Canvas app on Facebook, we load the Canvas URL within an `iframe` on that page. This results in your app being displayed within the standard Facebook chrome" (Facebook Developers n.d.).

[61] As described in the previous footnote, the Canvas URL points to the external host where the app is located which is then loaded within an iFrame in Facebook.

[62] Andreessen's examples of Level 3 Runtime Environment platforms include Salesforce which allows users to inject their own code and Andreessen's own Ning platform "for creating and running social networking applications" (2007b). Despite Andreessen's claim that all "platforms are good, period" he does state that "I call these Internet platform models *"levels"*, because as you go from Level 1 to Level 2 to Level 3, as I will explain, each kind of platform is *harder to build*, but much *better for the developer*." In this sense he promotes Level 3 platforms, including his own Ning, as being the 'best' platforms for developers.

[63] Level 3 "Runtime Environment" platforms are mostly located in the business-to-business domain such as SalesForce or Amazon.

635). The Extensible Markup Language (XML) addresses these issues by separating content, structure and presentation in a text-based format for machine consumption (W3C 2014b).[64] This machine-readable and human-readable format enables the sharing of structured information between otherwise incompatible systems (Myllymaki 2002, 635; W3C 2014b). XML has been an extremely important development for the web by making website data machine-readable and interchangeable between different systems. It enables the structured formatting of data for transmission and forms the basis for various data exchange mechanisms that let website data flow out and into other websites. [65]

According to Alan Liu, the separation of content and presentation informs the underlying technologic of the "post-industrial, transmission of information" which requires content be made "transformable", "autonomously mobile", and "automated" (Liu 2004, 57–58). This separation, so Liu continues, makes content "transcendental" so that it can be poured from one container into another, moving from database to database on the web (2004, 59). Liu describes how XML signals a shift from the first generation of self-contained HTML websites to a new type of website that is filled with content from external databases (2004, 57). These new web pages employ what Liu calls "data pours" to pull in and display dynamic content from third parties. A data pour is code embedded in a web page demarcating a space or container on that page that transfers data from and to external databases (2004, 59).

Published in the very early days of Web 2.0, Liu's idea of data pours can be read as an early reflection on the increasing modularity of the web, which he later updated as follows:

> My observations here about data pours apply with even more force in Web 2.0, where user-produced content flows both in and out of back-end databases through "template" Web pages that are often elegant, minimalist designs built around an all-powerful, blind aperture of parameterized

[64] The structure of an XML document looks as follows:

```
<book category="Fiction">
  <title lang="en">Emma</title>
  <author>Jane Austen</author>
  <year>1916</year>
</book>
```

[65] XML is at the core of several important data exchange mechanisms on the web, including XML-RPC and SOAP. The XML-RPC protocol is based on the idea of remote procedure calls (RPC) to "provide for transfer of control and data across a communication network" (Birrell and Nelson 1984, 39). It was developed in 1998 by Dave Winer from Userland and Microsoft to make requests to a remote computer and exchange data on the web (Laurent et al. 2001, x). Out of their work on XML-RPC came SOAP, Simple Object Access Protocol, a "lightweight protocol used to exchange XML-encoded information" (Laurent et al. 2001, 172). XML-RPC and SOAP-based web services enable the exchange of structured data between different machines on the web by communicating via the HTTP transmission protocol. XML and SOAP are technologies that have "formed a programmable web, one that extended the human web for the convenience of software programs" (Richardson and Ruby 2008, xviii)

Recently, JSON has become the preferred format over XML to transmit data, as it considered a more light-weight format. In addition, the architectural style REST, Representational State Transfer, has gained prominence for building web services. For example, social media platform Twitter offers a REST-based API which returns data in JSON.

> code—like a reversed black hole—that sucks all content in and throws it out again' (Liu, 2008, p. 320).

These characteristic data pours of Web 2.0, I argue, not only make a website's boundaries porous, but also establish continuous data flows with external databases. At the same time they set up data channels between websites and external databases.

The modularization of content and features

In separating content from presentation XML compartmentalizes web content by structurally describing each element on a web page and turning these into small modules of data that can be reused. The compartmentalization of content makes existing content available on the web and enables the commodification of content. Modularization is a key aspect of modern software design that enables the management of complex systems by dividing them up into smaller modules and the reuse of these modules (Baldwin and Clark 2000; Gehl 2012; McKelvey 2011). Within Web 2.0, Ullrich et al. argue, "services often disseminate their functionality by plug-in modular components, so called widgets" (Ullrich et al. 2008). These widgets enable the integration of a service's content and functionality into another website with a few lines of code that create a data pour. Widgets have become important platform-specific objects for social media platforms to distribute their content across different web spaces and to extend themselves into the web.

An important development towards this extension came from video sharing site YouTube. On July 7, 2005 YouTube announced a new feature that enabled users to put a list of their YouTube videos on their own websites by copy-pasting the provided HTML-code (YouTube 2005). This code embedded a YouTube widget showing a list of videos and thumbnails that linked to the videos on YouTube. A month later YouTube announced a new widget that embedded a video player so YouTube videos could now directly be played from within any website (YouTube 2005). The widget made it possible to distribute and view YouTube videos outside of YouTube's website. This video embedding feature is often seen as an important factor in the success of YouTube as it enabled the circulation of YouTube videos across social networks, blogs and other parts of the web by decentralizing platform features (Cheng, Dale, and Liu 2008).

While YouTube created its own widgets to distribute content *outside* of its website, social network MySpace played an important role in popularizing the role of third-party widgets to share content *inside* of its social network. In contrast to other social networks that were popular in 2005-2006—such as Friendster and later Facebook—MySpace allowed users to insert embed codes into their profile pages to add music players, photo albums and videos. It was the first social network that had such a rather open architecture and with it arose a culture of profile customizing and accessorizing (boyd 2007b).

With the ability to insert embed codes into profile pages third-party developers started to create little widgets to enhance the looks or functionality of MySpace. In November 2005, RockYou launched their first MySpace Flash widget to create and display photo slideshows. An important aspect of these early widgets is that, unlike YouTube's sharing widgets, they did not directly interface with MySpace's database. Users could not load their photos directly from MySpace into the widget because MySpace did not offer structured access to these photos. Instead, users had to upload their photos to external image hosting website ImageShack within the RockYou widget first (Tokuda 2009). This lack of directly interfacing with MySpace's database is seen in light of what Robert Gehl refers to as MySpace's "abstraction failure" to extract and monetize the content from its network (2012, 111–112).

This abstraction failure also distinguishes MySpace' and YouTube's widget approaches to distributing content. Whereas MySpace widgets were mostly oriented towards integrating and distributing content within its own network, YouTube's widgets were oriented towards the distribution of content and functionality outside of its network. As many Web 2.0 websites started to offer embed codes and widgets to distribute their content across the web, the approach of decentralizing platform features became the common widget approach of Web 2.0. A second important distinction is that, unlike MySpace widgets, YouTube widgets directly interfaced with the site's database. However, YouTube's database facing widgets were based on *one-way data streams*, where content is retrieved from the database and displayed on an external website. The next generation of widgets would be based on directly interfacing with databases to enable *two-way data streams* to not only read data from the database but also to write new data to it.

Interfacing with databases

Facebook's Social Plugins are a set of plugins, or widgets, including the ubiquitous Like button "that let you share your experience off of Facebook with your friends and others on Facebook."[66] The plugins function as platform modules to extend platform functionality into external websites (cf. Bodle 2011, 325–326). At the same time, Taina Bucher argues, they function as "edge-creating devices," which collect data created by connections or "edges" off Facebook.com and send it back to the platform's databases (2012a, 6). Social plugins are an important part of Facebook's platform architecture to enable the decentralization of platform functionality and data and the recentralization of data produced outside of the platform (see chapter 4). By embedding a plugin into their website webmasters set up two-way data channels in which data continuously flows between the site and Facebook's database. However, before these plugins can interface with Facebook's database from an external

[66] See: https://www.facebook.com/help/103828869708800 [Accessed 2 February 2015].

website, webmasters need to make their websites compatible with Facebook's platform infrastructure.

To integrate social plugins webmasters need to embed a piece of JavaScript code into their websites which sets up a data communication channel with Facebook's platform. This code initiates the Facebook Software Development Kit (SDK)[67] for using Social Plugins, Facebook Login and making API calls to the database. In doing so, webmasters are making their pages platform ready for continuous data flows with Facebook. This notion of making external websites and web data "platform ready" extends Tarleton Gillespie's idea of how data is made "algorithm ready" (2014, 168) to highlight the role of the platform infrastructure in reconfiguring external data to fit the agenda of the platform.

Another important part of Facebook's platform infrastructure is the Open Graph, which is explicitly geared towards making external data platform ready. The Open Graph "lets you integrate apps deeply into the Facebook experience, which increases engagement, distribution and growth" (Facebook Developers 2015c). To integrate an app, developers need to use the Facebook SDK and Facebook Login to set up relations between the app, Facebook and the user (Facebook Developers 2015c). This integration lets apps tell 'stories' on Facebook such as "Mary ran 6 miles with MyRunningApp" (Facebook Developers 2015c). Apps submit these stories to the Open Graph in a very structured manner organized around four elements, e.g.: John (actor) is reading (action) The Odyssey (the object) on Goodreads (app). There are a number of predefined actions such as 'like', 'watch' and 'read' but developers can also create their own. Taina Bucher describes these efforts from Facebook "as a way to build a semantic map of the Internet" (Bucher 2012a, 5). The app integrations enable Facebook to collect external app data and activities in a very structured manner, send it back to the database and connect it to a user and other data. It further expands Facebook's data collection techniques into external applications and formats this data according to the logic of the platform so it can be put into new relations within the platform.

Webmasters can also make their websites platform ready by marking up their sites with Open Graph tags. These meta tags provide Facebook's crawler with "structured info about the page such as the title, description, preview image, and more" and control how content appears on Facebook to "improve distribution and engagement" (Facebook Developers 2015a). Similar to the practices of webmasters optimizing their pages for search engines (see chapter 4), these practices can be seen as a form of social media optimization.

The Open Graph shows how Facebook strictly formats data flowing from apps and external websites to the platform in order to make it platform ready. Whilst platforms position themselves as neutral intermediaries (Gillespie 2010, 252) or as neutral "utilities transmitting communication and information data" (Van Dijck 2013c, 6) they do not only actively shape

[67] See: https://developers.facebook.com/docs/javascript/quickstart/v2.2 [Accessed 2 February 2015].

and intervene in social activities (Gillespie 2015, 1), but they also format data passing through their infrastructure according to the logic of their underlying infrastructures.

The dual logic of platformization

These two examples show how Facebook employs its platform as an infrastructural model to extend itself into external online spaces and as an economic model to employ these extensions to reformat data for its platform. I have introduced the notion of platformization to understand these processes from an infrastructural perspective. The platformization of the web refers to the rise of the platform as the dominant infrastructural and economic model of the social web and the consequences of the expansion of social media platforms into other spaces online. Platformization, I argue, rests on the dual logic of social media platforms' expansion into the rest of the web and, simultaneously, their drive to make external web data platform ready. As an infrastructural model, social media platforms provide a technological framework for others to build on which, I argue, is geared towards connecting to and thriving on other websites, apps and their data. Making external data amenable for their own databases is, so I suggest, central to the economic model of social media platforms.

These two processes of decentralizing platform features and recentralizing platform ready data characterize what I call the double logic of platformization. This double logic is operationalized through platform-native objects such as APIs, social plugins and the Open Graph, which connect the infrastructural model of the platform to its economic model. These platform-native objects serve as prime devices for social media platforms to expand into the web and to create data channels—data pours—for collecting and formatting external web data to fit the underlying logic of the platform. That is, I have shown how social media platforms are enacting their programmability to reweave the web for social media. This process, I conclude, also requires us to revisit the critiques of social media platforms as walled gardens.

From walled garden critique to platformization critique

Facebook's Social Plugins show how social media platforms present a different architectural model than social networks sites. Widgets decentralize platform features into the web which do not create small openings in the walled garden, but instead create data pours in which data continuously flows. The Facebook Comments plugin, for example, creates a Facebook-enabled comment space on an external website which users can comment on with their Facebook account. These comments are sent back to the platform's database. Depending on the user settings, these comments may also be posted to the user's News Feed within Facebook which enables the conversation to continue there. The plugin functions as a data pour to exchange comments between external websites, Facebook's database and Facebook's Newsfeed. As such, it may create a conversation between a user inside of Facebook and a user on an external website. Such mechanisms challenge our conception of Facebook as a walled garden.

Social Plugins set up channels for continuous data flows and data production.[68] This may be seen in the way they turn the declaration of interests on the social network profile, as a form of taste performance (Liu 2007), into a software-assisted, continuous and distributed process within social media platforms. On social network sites such as MySpace users could fill in or select their interests on their user page. Within social media platform Facebook this has turned into an automated process since interests are automatically derived from liked objects. These objects can either be located within or outside of Facebook, thereby turning the declaration of interests into a distributed process. In addition, it can be seen as a continuous declaration of interests as one is never finished liking.

These examples show how social media platforms through their expansion into external websites and apps enact a different architectural than social network sites. This also informs their economic model which is based on a carefully regulated extension to produce more data for the platform outside of its own boundaries (Bodle 2011) (see chapter 5). By letting data flow out, platforms enable more engagement with their data and set up data channels to integrate continuous data flows into them. Social media platforms invite their data to circulate outside of their boundaries but only, Taina Bucher argues, under the conditions of these platforms in which the platform's API acts as a powerful governing technique (2013).

I have argued that the API, which enacts a platform's programmability, also connects the infrastructural model of social media platforms to their economic model. Whereas the business strategy of social network sites was focused on the commodification of user content and activities within their walled gardens by pulling users and content in, social media platforms face outwards and integrate themselves into other web space to make external data valuable for their platforms. In this chapter I have argued that with the shift from social network *sites* to social media *platforms* the walled gardens metaphor is hard to maintain which requires us to revise our conceptual frameworks for understanding and critiquing these sites. Therefore, I have put forward the notion of platformization, which pays close attention the medium-specific characteristics of platforms in the computational sense.

As a contribution to platform studies and social media studies, platformization focuses on the material-technical infrastructures of platforms and shows the work that social media platforms do in a computational sense. Such perspective demonstrates how social media platforms do not only function rhetorically but also enact a very specific technological infrastructure and economic business model (cf Gillespie 2010). Platformization foregrounds the programmability of platforms and draws our attention to examining data pours and their politics. It allows for analyzing the changing politics of data exchanges in the social web in which platforms reformat external web data before it enters their platform. Further research

[68] Using cookies, Social Plugins initiate even more continuous data flows by sending information about the website's visit back to the platform.

should examine other mechanisms that platforms employ to extend themselves into the web and to make data platform ready (see chapter 7). This has been an exploration in that area showing how platformization can be seen as a mechanism by which social media platforms are turning the web more and more into social media. The next chapter explores how social media platforms have expanded into the blogosphere through bloggers' adaptation of widgets to integrate social media platform data and functionality into blogs thereby connecting blogs and social media platforms not through traditional hyperlinks but through platform features.

3. The coming of the platforms: Rethinking the history of the Dutch blogosphere

The blogosphere has played an instrumental role in the transition and evolution of linking technologies and practices, such as the introduction and development of the permalink, trackback and pingback[69] and their use by bloggers to develop a culture of blogging as a distinct online culture.[70] Important research in this area has been practice, event or issue based, trying to capture an otherwise fleeting phenomenon in real-time, before it is deleted, overwritten or no longer available. Now that blogging has reached maturity the first historical accounts are being created. This chapter provides such a historical account of the Dutch blogosphere in relation to the rise of social media. In particular, I focus on the changing link structure of the blogosphere and bloggers' adaptation of sidebar widgets, which do not connect blogs by conventional hyperlinks but rather by social media features.

A 2010 study by Pew Internet claimed that blogging is in decline because young adults make use of social media platforms such as Facebook and Twitter instead of running their own blogs (Zickuhr 2010). This study about the supposed decline of blogging led to a great number of articles on technology blogs and in the trade press with attention-drawing titles such as 'Blogs Wane as the Young Drift to Sites Like Twitter' in *The New York Times* (Kopytoff 2011) and 'The End of Blogging' in *The New York Observer* (Duray 2011).[71]

In 2013, longtime blogger Jason Kottke reflected on the changing status of blogging by describing how "instead of blogging, people are posting to Tumblr, tweeting, pinning things to their board, posting to Reddit, Snapchatting, updating Facebook statuses, Instagramming, and publishing on Medium" (2013). Kottke argues that social media platforms have taken over some of the core functionality of blogging since "[t]he primary mode for the distribution of links has moved from the loosely connected network of blogs to tightly integrated services like Facebook and Twitter" (2013). Other professional bloggers such as Mathew Ingram have described this shift not as a decline in blogging or the end of blogging but rather as an evolution of the practice (2011). This chapter engages with such ideas about the evolution of blogs, the practice of blogging and the blogosphere with an empirical case study examining the changing linking practices in the Dutch blogosphere with the rise of social media and the

[69] The next chapter provides a detailed account of how different actors in the blogosphere have contributed to the development of new types of links.

[70] This chapter has been previously published as: Weltevrede, Esther, and Anne Helmond. 2012. "Where Do Bloggers Blog? Platform Transitions within the Historical Dutch Blogosphere." *First Monday* 17 (2). doi:10.5210/fm.v17i2.3775. See acknowledgements on page iii.

[71] Such articles titles about "the death of blogging" are often considered link-bait, a term referring to catchy titles that draw attention and encourage links back to the article.

integration of social media platform content. In order to understand the changes in the medium of the blog, the practice of blogging and the structure of the blogosphere, we need to turn to social media. The aim of this chapter is to trace the changing link structure and definition of the Dutch blogosphere with the rise of social media.

In doing so I seek to contribute to the growing body of literature engaging with historical accounts of the blogosphere by investigating structural changes in the blogosphere's underlying infrastructure and bloggers' linking practices over time. More specifically, I seek to contribute to empirical research into historical blogospheres and hyperlink analysis by proposing methods to 1) reconstruct historical blogospheres using the Internet Archive Wayback Machine, 2) explore the rise of social media platforms in the historical Dutch blogosphere 3) examine how social media has modified the blogosphere's link structure and 4) redefine what is considered an actor in the blogosphere.

This chapter addresses both methodological questions related to the empirical research of historical blogospheres and presents the outcome of research into the first phase of the platformization of the Dutch blogosphere.[72] This research not only provides a model for studying historical blogospheres but also presents methods to study the platformization of the blogosphere. As the first known empirical study into the history of the Dutch blogosphere, it offers insights into the changing shape of the Dutch blogosphere and its interconnections with the rise of social media.

Studying contemporary and historical blogospheres

The blogosphere is often studied by mapping and visualizing the interconnections between blogs, in order to make the blogosphere tangible and visible (Adamic and Glance 2005; Bruns 2007; Hurst 2007; Kelly and Etling 2008). In other words, to become visible, the image of the blogosphere must be constructed, either by blogosphere related services such as directories, web rings and blog search engines or by academic network visualizations. Such network visualizations may be created by employing web crawlers for network analysis that crawl websites and capture the outlinks from these sites. The techniques of crawling and network analysis have been used for various analytical purposes such as tracking conversation patterns in the blogosphere by using the IssueCrawler tool[73] to detect issue networks (Bruns 2007), crawling the front pages of blogs to reflect blogroll communities (Adamic and Glance 2005), and large scale grouping of linked blogs to define clusters of shared informational worlds (Kelly and Etling 2008). Although these different tools and methods produce different

[72] The project page of this empirical research including data, tools and visualizations is located at http://dutchblogosphere.digitalmethods.net [Accessed 11 March 2014]. As stated in the acknowledgements this research project has been conducted together with colleague Esther Weltevrede.

[73] A software tool locating and visualizing networks on the web, developed by the Govcom.org Foundation, Amsterdam. See: http://issuecrawler.net [Accessed 2 February 2012].

network visualizations, they all provide graphical representations of interconnections and insights into the overall structure of the blogosphere and its actors (Highfield 2009). In this chapter I will demonstrate that choices in method do not only shape the blogosphere but *also* the definition of the blogosphere and blogs. I will show that the blogosphere can no longer be seen as a distinct network created through the interconnections between blogs but that the blogosphere has become entangled with the social media ecosystem.

Turning to historical blogosphere analysis this body of research mainly consists of ethnographic research providing personal stories and anecdotes (Blood 2004; Rosenberg 2010). In addition, there is also a small body of empirical work, which includes research on the structure and evolution of the LiveJournal blogspace[74] (Kumar et al. 2004), the birth of the U.S. blogosphere (Ammann 2009) and the rise of the early A-list blogosphere (Stevenson 2010a). In one of the earliest historical blogosphere studies authors Kumar et al. suggest a method to map a blogspace—a blogosphere—over time by making use of the blog's own archival capacities (2004). The researchers employ a crawler to collect data from a set of active blogs by fetching current and archived blog post entries to analyze changing linking patterns and interest clustering over time (2004, 36). In order to do so, they make use of features in blog entries such as time stamps and links as well as features in profile pages such as interests and demographics. As a consequence of working with an active set of blogs, this method is specifically suitable to analyze changes in an existing blogosphere over time. The question arises how we can study changing linking patterns in blogs retrospectively. As blogs come and go, the objects of analysis may go offline, which makes it hard to study changes in the links between them. In this chapter I propose a method to do historical hyperlink analysis by making use of web archives to examine structural changes in the blogosphere over time.

Existing web archives such as the Internet Archive lend themselves to studying previous states of the web because they create time-stamped snapshots of websites and blogs. In particular the Wayback Machine—the interface to the Internet Archive's web archives—provides a valuable source for web historians because of its accessibility and scope. The archive contains over 456,000,000,000 URLs from late 1996 to up to a week ago and provides snapshots from a wide range of websites.[75] Although the interface of the Internet Archive Wayback Machine foregrounds "website biographies" or "single-site histories" (Rogers 2013, 66)—as only single URLs can be retrieved—the archived snapshots may be used in a variety of ways (see also chapter 6). For example, Rudolf Ammann (2009) uses the Internet Archive to study the emerging blogosphere by mapping linking patterns of early blogs, while Michael Stevenson (2010a) outlines a method to re-purpose the Internet Archive by using the early blog index EatonWeb as a historical resource to create a custom archive. In a next step

[74] Kumar et al. use the term 'blogspace' instead of the more commonly used term 'blogosphere' to refer to "the collection of blogs and all their links" (2004, 35). In their research they focus on the blogspace of LiveJournal.

[75] See: https://archive.org/web/ [Accessed 23 March 2015], https://blog.archive.org/2013/10/25/fixing-broken-links/ [Accessed 17 March 2014] and http://blog.archive.org/2013/01/09/updated-Wayback/ [Accessed 17 March 2014].

Stevenson recreates the blogosphere by "conjuring up" blogs that have not been archived by means of historical network analysis (2010a).

The research presented here builds on the above-mentioned methods and tools by developing a number of novel methods and techniques that focus on infrastructural changes in the blogosphere. In addition, it makes a contribution to historical blogosphere analysis and social media studies by focusing on the tight relationship between social media platforms and blogs, which has remained understudied so far. I wish to address two contributions that have been made in this area. First, in his analysis of the social media ecology of the 2010 Toronto G20 protests Thomas Poell examined the hyperlink networks of protest communication and found that these networks form a hybrid ecology of blogs, social media platforms, media sharing sites, news websites and other types of sites (2013). Second, in her revised edition of *Blogging* Jill Walker Rettberg has included a section on blogs' adaptations to a social media ecosystem (2013, 14–17) in which she argues that conversations in the blogosphere are no longer confined to blogs but also take place on social media platforms such as Facebook and Twitter (2013, 14–15).

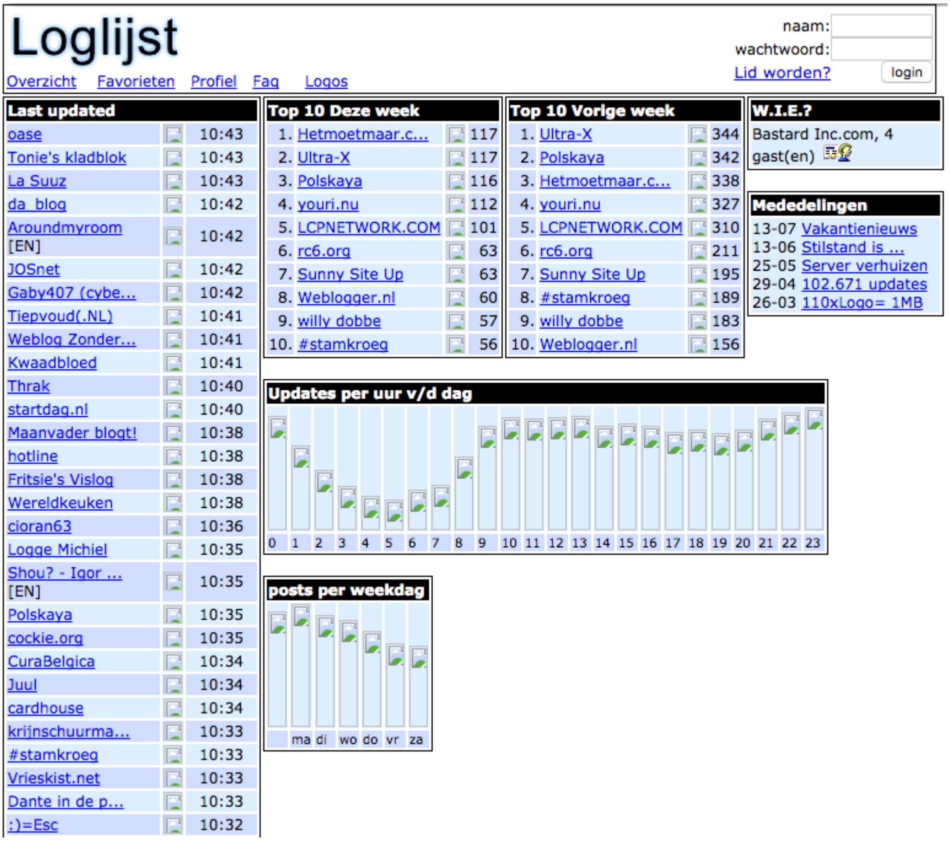
However, empirical approaches that investigate the relations between blogs and social media platforms remain scarce. To address this, I will propose a method for a fine-grained analysis of the link networks created between blogs and social media through widgets. I consider this as a first step towards studying the platformization of the Dutch blogosphere, that is, how social media platforms have extended into blogs and have woven themselves into the blogosphere.

Retrieving Dutch blogs

In order to analyze infrastructural changes in the Dutch blogosphere over time we need to construct historical snapshots. In a first step to reconstruct the historical Dutch blogosphere I created a corpus of Dutch blogs. I located and retrieved a collection of blogs from a 2001 database dump—containing 631 unique blogs—from *Loglijst* (see figure 5), an early Dutch blogosphere indexing initiative.[76] In addition to these blogs, I compiled expert lists mentioning Dutch blogs from interviews, books and authoritative lists found on the web and in the Internet Archive's Wayback Machine. These experts lists include long list nominations for the Dutch blog awards, the Dutch Bloggies from 2001-2008, all blogs mentioned in two seminal pieces on the history of the Dutch blogosphere by Dutch blogosphere historians Frank Schaap (2004) and Frank Meeuwsen (2010) and finally a list citing "Weblogs that really

[76] Loglijst was a blog directory similar to EatonWeb and Technorati which focused on Dutch and Flemish blogs. I received an old database backup from the people behind Loglijst, Jan–Willem Hiddink and Robert–Reinder Nederhoed at Frank Meeuwen's *Bloghelden* (2010), meaning 'blogging heroes' in Dutch, book launch where I presented early ideas from this research on the historical Dutch blogosphere. My colleague Erik Borra created a new custom database from the old Loglijst database file for this research.

matter” in a December 2010 blog post by Bert Brussen,[77] former blogger for the famous Dutch “shocklog” *Geenstijl*.[78] Relying on these sources to provide me with a collection of Dutch blogs led to include a small number of Belgian (Dutch language) blogs that these expert sources considered being part of the Dutch blogosphere.[79] The corpus, collected from these expert lists, amounted to 2507 blog URLs that served as the starting points for the next part of the analysis. [80]



Loglijst

Overzicht Favorieten Profiel Faq Logos

naam:
wachtwoord:
Lid worden? login

Last updated	
oase	10:43
Tonie's kladblok	10:43
La Suuz	10:43
da_blog	10:42
Aroundmyroom [EN]	10:42
JOSnet	10:42
Gaby407 (cybe...	10:42
Tiepvoud(.NL)	10:41
Weblog Zonder...	10:41
Kwaadbloed	10:41
Thrak	10:40
startdag.nl	10:40
Maanvader blogt!	10:38
hotline	10:38
Fritsie's Vislog	10:38
Wereldkeuken	10:38
cioran63	10:36
Logge Michiel	10:35
Shou? - Igor ... [EN]	10:35
Polskaya	10:35
cockie.org	10:35
CuraBelgica	10:34
Juul	10:34
cardhouse	10:34
krijnschuurma...	10:33
#stamkroeg	10:33
Vrieskist.net	10:33
Dante in de p...	10:33
:)=Esc	10:32

Top 10 Deze week	
1. Hetmoetmaar.c...	117
2. Ultra-X	117
3. Polskaya	116
4. youri.nu	112
5. LCPNETWORK.COM	101
6. rc6.org	63
7. Sunny Site Up	63
8. Weblogger.nl	60
9. willy dobbe	57
10. #stamkroeg	56

Top 10 Vorige week	
1. Ultra-X	344
2. Polskaya	342
3. Hetmoetmaar.c...	338
4. youri.nu	327
5. LCPNETWORK.COM	310
6. rc6.org	211
7. Sunny Site Up	195
8. #stamkroeg	189
9. willy dobbe	183
10. Weblogger.nl	156

W.I.E.?

Bastard Inc.com, 4 gast(en)

Mededelingen	
13-07	Vakantienieuws
13-06	Stilstand is ...
25-05	Server verhuizen
29-04	102.671 updates
26-03	110xLogo= 1MB

Updates per uur v/d dag

posts per weekdag

Figure 5: Loglijst, an early Dutch blogosphere indexing initiative. The site shows the last updated Dutch blogs, the top 10 Dutch blogs from this week and last week, the updates per hour and the number of blog posts per day in the Dutch blogosphere. Screenshot from the Loglijst website as archived in the Internet Archive’s Wayback Machine, 24 July 2002. Retrieved from: http://web.archive.org/web/20020724165825/http://www.nederhoed.com/~jeewee/ [Accessed 1 September 2011].

[77] See: http://www.dejaap.nl/2010/12/28/verplicht-in-uw-rss-reader-Weblogs-die-er-echt-toe-doen/ [Accessed 1 September 2011].

[78] Bert Brussen has left GeenStijl to start a Dutch group blog focused on current events called The Post Online, available at http://www.thepostonline.nl/ [Accessed 14 March 2014].

[79] Belgium borders the Netherlands and the two countries share a common language: Dutch (Flemish).

[80] The full list can be downloaded from: https://dutchblogosphere.digitalmethods.net/data/StartingPoints_Dutchblogosphere.csv [Accessed 18 March 2014.

I then turned to the Internet Archive's Wayback Machine to locate and retrieve archived snapshots of these blogs. I queried the Wayback Machine for each blog's URL between 1999 and 2009 using the Internet Archive Wayback Machine Network Per Year tool.[81] This tool selects the result dated closest to the middle of each year under investigation. From the 2507 URLs requested I was able to retrieve 946 blogs from the Internet Archive. This method yielded a custom collection of archived copies of historical Dutch blogs for each year between 1999 and 2009 with a timestamp near the middle of the year. Only blogs with a copy in the Internet Archive were retained for further analysis. The following table 1 represents the number of blogs per year serving as starting points for further analysis:

1999	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009
24	138	456	816	778	863	850	788	717	860	723

Table 1: The number of archived blogs from the expert lists that were retrieved from the Internet Archive Wayback Machine per year between 1999 and 2009. These archived blogs serve as the starting points for further analysis. The URLs were retrieved by the Internet Archive Wayback Machine Network Per Year tool.

In the following section I propose a method to create yearly network views, historical blogospheres, from this custom collection of archived blogs. Mapping the interconnections between blogs over time allows for analyzing the changing link structure of the Dutch blogosphere and for tracing the rise of social media platforms.

Reconstructing the blogosphere

In 1999, Brad L. Graham coined the term 'blogosphere' to mark the end of cyberspace: "Goodbye, cyberspace! Hello, blogiverse! Blogosphere? Blogmos?" (1999). William Quick revived the word as "the intellectual cyberspace we bloggers occupy" and explicitly stated that the blogosphere is a space for serious discourse (2002). danah boyd similarly described the blogosphere as "the imagined public sphere," (boyd 2006) echoing the idea of the blogosphere as a discursive space. One of the first bloggers, Dave Winer, defines a blog as "The unedited voice of a person" (2003). Others, such as Geert Lovink, position blogs as a reaction to mainstream media (2008). Besides the notion of the blogosphere as a space for discourse, other definitions stress the formalistic characteristics of the blogosphere as an interlinked set of blogs which "allows for the networked, decentralised, distributed discussion and deliberation on a wide range of topics" (Bruns, Kirchhoff, and Nicolai 2009). A complimentary approach to the blogosphere as an interlinked set of blogs looks at how blogs are:

[81] See: https://tools.digitalmethods.net/beta/waybackNetworkPerYear/ [Accessed 12 April 2014].

> embedded into a much bigger picture: a segmented and independent public that dynamically evolves and functions according to its own rules and with ever-changing protagonists, a network also known as the 'blogosphere'. A single weblog is embedded into this network through its trackbacks, the usage of hyperlinks as well as its so-called "blogroll" – a blogosphere-internal referencing system" (Bross et al. 2010).

Further extending this line of thinking, i.e. blogs are embedded in a larger networked ecology with shifting actors, the blogosphere may also be defined by including the actors they link to in their networked ecology:

> The notion of a mini-blogsphere additionally rests on the extent to which the set of blogs doing an issue are interconnected by links and/or by textual referencing. Blogs also make [sic] be 'connected' together through common references to a third-party, *e.g.*, all blogs linking to or referencing a particular piece in the *New York Times*" (Rogers 2005).

In a formal sense the blogosphere can be defined either as an independent network of blogs created by the interconnections between blogs (Herring et al. 2005; Bruns, Kirchhoff, and Nicolai 2009; Bross et al. 2010) or as an interlinked network created by blogs linking to other blogs as well as other actors on the web (Rogers 2005; Chia 2012).

At first glance my approach to define and reconstruct a blogosphere presented in this chapter may appear formalistic because my definition of the blogosphere follows from the outlined method based on link analysis (see below). However, this definition follows from the cultural practices of bloggers, their blogging practices. Benkler and Shaw contend that "the term 'blogging' has more of a cultural meaning than a technical meaning" since the many different blog services, software types and available plugins permit the blog's custom use (2010, 13). In this chapter I show that the integration of social media widgets into blogs enable custom blogging practices but that their technical affordances also inform the definition of the blogosphere as created through interlinking.

In a next step I created annual snapshots of the Dutch blogosphere from the custom collection of archived blogs that were retrieved from the Internet Archive Wayback Machine. One of the consequences of working with this custom collection is that only research on front-page level and not on a post level is possible. The custom collection was created by requesting the host [blog.com] from the Internet Archive Wayback Machine which results in archived copies from the blog's front page. Hence this method may be viewed as a more structural blogosphere analysis rather than an issue (Bruns 2007) or event (Adamic and Glance 2005) analysis. It further expands Adamic and Glance's approach who studied the 2004 U.S. Election by creating snapshots of political blogs by capturing outlinks from the blogroll (2005, 36). Since blogrolls contain links to other bloggers and because bloggers do not change these links on a very regular basis they can be used to present "a more static picture of a broader blogosphere" (Adamic and Glance 2005, 36). The method presented here is not restricted to the blogroll to capture other structural actors in the blogosphere.

Although fully aware that the choice of starting points shapes the Dutch blogosphere, the methodology used in this chapter only retains blogs deemed relevant by other blogs. The proposed method to create a structural blogosphere is inspired by co-link analysis as used by the IssueCrawler, a software tool locating and visualizing networks on the web.[82] This co-link analysis is performed on the collection of blogs for each year between 1999 and 2009 in two steps: first, for each archived blog all links on front-page level are extracted (one depth) and subsequently, in the network visualization and analysis tool Gephi,[83] only nodes receiving at least two links from the starting points are maintained in the network visualization (one iteration).

The resulting network maps per year thus retain only co-linked actors, those receiving at least two links from the starting points. This implies that the starting points themselves may drop off the map and that new blogs may appear if at least two blogs from the starting points link to them. The approach acts as a validation of the expert lists because a blog needs to receive at least two links from other blogs. In addition, it also expands the expert lists by introducing new blogs that have been found relevant by other bloggers because they receive at least two links.[84] This method is a contribution to historical hyperlink analysis because it allows for "conjuring up" (Stevenson 2010a; Rogers 2013, 10) historical blogs that have not been archived, but which have been located and included in the network graph of the historical blogosphere by following the outlinks from the archived blogs. Table 2 shows the number of blogs that are included in the network graphs per year as a result:

1999	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009
4	241	785	1204	1434	1693	1732	1823	2103	2110	2190

Table 2: The number of blogs in the network graphs of the historical Dutch blogosphere per year between 1999 and 2009. The blogs from the expert lists (see table 1) were retrieved by the Internet Archive Wayback Machine Network Per Year tool. Subsequently, the outlinks of these archived blogs were followed and co-link analysis was performed.

Whereas co-link analysis is an analysis module most successful for locating issue networks, in this case, the result of the co-link analysis is that issue or event-based links are excluded from analysis. This has three reasons: first, the starting points have not been chosen because they share an issue or an interest in an event, but because they share the practice of blogging in the Dutch web space. Second, only front pages are crawled, which means that the more structural links are followed, such as links in blogrolls, and links to blog-related services and social media platforms. In other words, these links are the stable variable in the analysis, whereas links in posts are only taken into account if present on the front page. Third, the time frame of each network is one year. This method therefore excludes links to versatile issues dominating the Dutch blogosphere for a short period of time and focuses on the more structural linking practices.

[82] See: http://issuecrawler.net [Accessed 2 February 2012].

[83] See: https://gephi.github.io/ [Accessed 1 May 2015].

[84] In addition to the 946 starting points, the method has led to include 6768 new URLs.

In what follows, I further describe how I constructed the Dutch blogosphere by using archived snapshots from the Internet Archive's Wayback Machine and how I prepared them for further analysis. Specific attention is paid to process of constructing the blogosphere by reconfiguring actor definitions—to detect the actors that create the blogosphere—and reconsidering interlinking practices. This approach gives novel insights into the composition of the blogosphere and its actors. This method consists of two strands: first I refine the network analysis by defining the actors using Gephi and G-Atlas software,[85] and then I complement the network analysis by color-coding the social media platforms present in the blogosphere.

Defining the actors

As previously described, I retrieved snapshots of the blogs in the collection between 1999 and 2009 from the Internet Archive and extracted their outlinks on a front-page level and put the results in Gephi, a tool for visualizing and analyzing network graphs.[86] In Gephi, I performed a simplified version of the IssueCrawler's co-link analysis so that only blogs with more than two links from our starting list are maintained. Co-link is performed on a "by site" level, as it is more indulgent than the "by page" option because it counts all links from site to site. In other words, co-link analysis is performed on the hosts and not on the deep pages.

A common problem in online network visualizations is that big domain nodes (e.g. twitter.com or blogspot.com) take a prominent position in the graph (Shaw and Benkler 2012, 463). Analysis of these maps often suggests that the debate is moving elsewhere (i.e. to social media). In an attempt to untangle the big social media platform nodes in the Dutch blogosphere, I propose to redefine the nodes of the network to actors. Most network analysis software treats the host and in some cases sub-host as the actor. However, in this case the "actor" or blogger is often defined after the slash, like the early bloggers that started blogging from their personal homepage (e.g. xs4all.nl/~zweers) or the recent microbloggers on Twitter (e.g. twitter.com/2525). A similar approach has been developed by researchers from Médialab Science Po who have defined the concept of "web entities" to unravel pages grouped by domain name (Girard 2011). Also Benkler and Shaw, in their work on the U.S. political blogosphere, stress the importance to analyze what is *inside* the large network nodes in order to specify their internal differences (2010).

To identify nodes in the blogosphere as actors, I redefined "actors" on a URL level. This requires an additional step in the analysis because not all URLs follow the same pattern. With most web sites "actor" equals "host" (e.g. example.com) while actors on hosted blog software

[85] Gephi is open source software for visualizing and analyzing large networks graphs, located at http://gephi.org and the G-Atlas software is developed by TIC Migrations in Paris, located at http://ticmigrations.fr/ [Accessed 4 February 2012].
[86] Ibid.

services are defined before the host on a subdomain (e.g. example.blogger.com), actors on personal homepages are often defined by their ~ after the slash (e.g. xs4all.nl/~example) just like microbloggers on Twitter are defined after the slash (e.g. twitter.com/example). With the use of Google Refine, “a power tool for working with messy data”[87] I “coded” each of the actors in GREL (Google Refine Expression Language) to automatically search, transform and count the actors in the network (see Appendix A). In this actor definition project I sought to formalize “URL patterns” in the network graphs to analyze the link structure of the blogosphere and the changing linking practices of bloggers in more detail.

The Dutch blogosphere in transition

As previously addressed, mapping the outlinks of the blogs retrieved from the Internet Archive between 1999 and 2009 allows for “conjuring up” the blogosphere and to go back in time and study how and where the Dutch blogosphere originated. Using the fine-grained actor definition, I visualized the network with Gephi for each year. Figure 6 shows the rise, evolution and first signs of decline of the Dutch blogosphere. In this figure grey depicts the hyperlink network of all the years together and red shows the blogosphere of a particular year. The first Dutch bloggers that appear on the network in mid 1999 are not interlinked into a “sphere”, so we can trace the beginning of a structural Dutch blogosphere back to 2000.

In 1999 the map (not displayed)[88] of the structural historical blogosphere only shows four nodes which are not linking to each other but which are present because they receive at least two links from our selected starting points. The four nodes are Nedstatbasic, Nedstat, *Wired* and a Dutch blog by Wessel Zweers, a.k.a. ~wzweers. Nedstat and Nedstatbasic are two related Dutch statistics providers. A familiar node is *Wired*, a technology magazine also prominent in the American early blogosphere (Stevenson 2010a). The only Dutch blogger of the four, ~wzweers, is hosted on one of the oldest Dutch hosting services providing free personal homepages, “De Digitale Stad” (DDS, Digital City). Well-known Dutch blogs from that period, like Sikkema, Prolific and Alt0169 (Meeuwsen 2010) are notably absent because they do not receive two links from the starting list’s blogs.

[87] OpenRefine (formerly Google Refine), see: http://openrefine.org/ [Accessed 1 May 2015].
[88] See: https://dutchblogosphere.digitalmethods.net/gatlas/resources/flash/GexfWalker.swf?gexfPath=../../engine/data/graph/17/attr5.gexf [Accessed 1 May 2015].

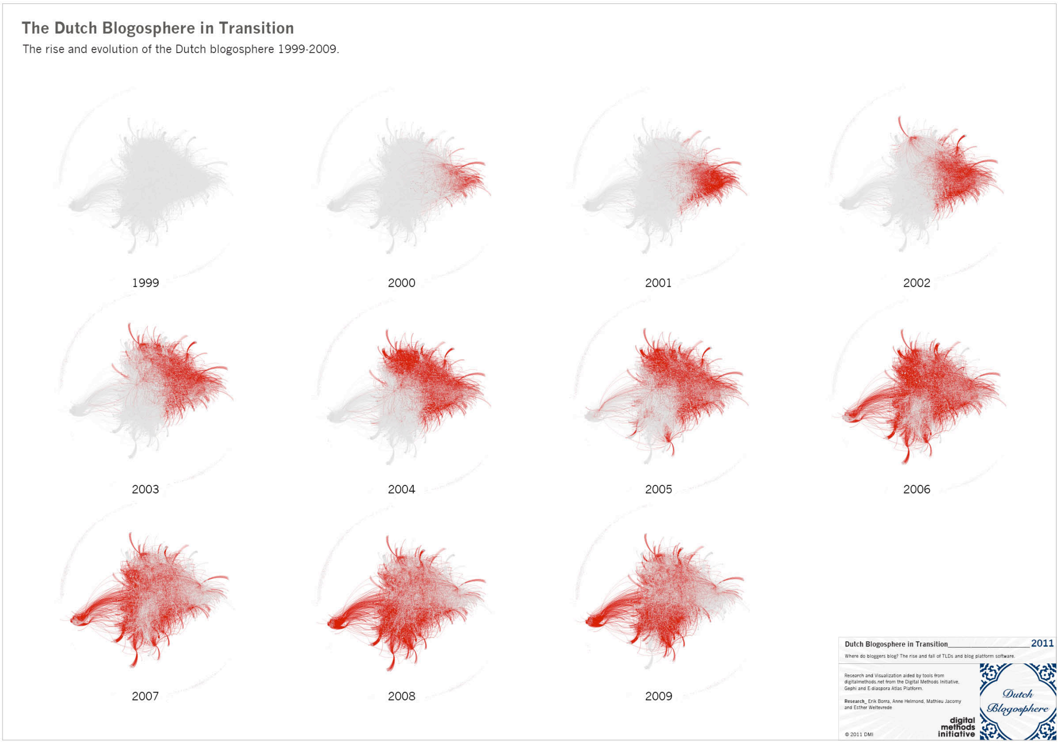

The Dutch Blogosphere in Transition
The rise and evolution of the Dutch blogosphere 1999-2009.
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
Dutch Blogosphere in Transition
2011
Where do bloggers blog? The rise and fall of TLDs and blog platform software.
Research and Visualization aided by tools from digitalmethods.net from the Digital Methods Initiative, Gephi and E-diaspora Atlas Platform.
Research_ Erik Borra, Anne Helmond, Mathieu Jacomy and Esther Weltevrede
© 2011 DMI
digital methods initiative
Dutch Blogosphere

Figure 6: The Dutch blogosphere in transition. The rise and evolution of the Dutch blogosphere between 1999 and 2009. Grey depicts the hyperlink network of all the years together and red shows the blogosphere of a particular year. The Dutch blogs were retrieved from the Internet Archive Wayback Machine between 1999 and 2009 using the Internet Archive Wayback Machine Network Per Year which outputs a network per year. Subsequently, the historical hyperlink networks were created with Gephi, based on a co-link analysis. Visualization created by Anne Helmond and Esther Weltevrede with Gephi and in Adobe Illustrator, 2011.

In order to further examine this pre-blogosphere I removed the co-link analysis (see figure 7) to explore the link structure of the 1999 blogs. Figure 7 shows that some of the well-known Dutch bloggers, as mentioned in Meeuwsen's *Blog Helden* (2010), together with less well-known bloggers, are present but do not form a blogosphere yet. Most notably Alt0169, ~wzweers and ~onnoz reach out to other Dutch blogs which can be seen as an effort to establish a community between blogs. The early Dutch bloggers are linking outward but do not receive reciprocal links to connect these loose blog networks into an interconnected blogosphere.

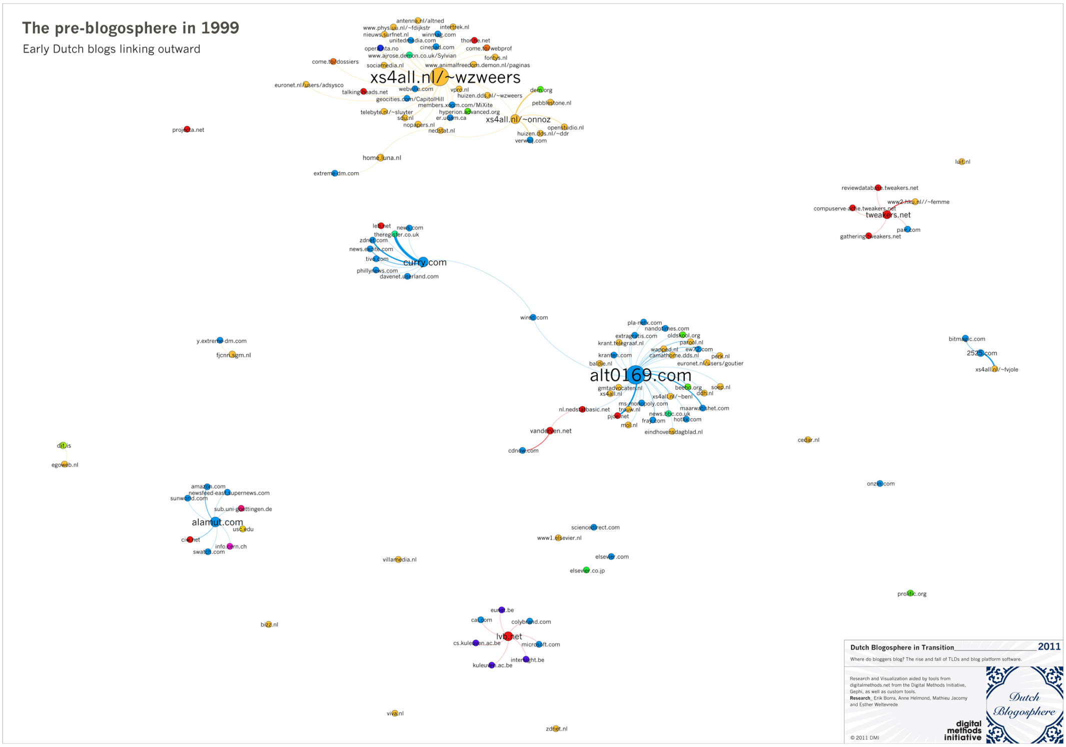


Figure 7: The pre-blogosphere in 1999 showing early blogs linking outward. The network graph displays the connections between blogs in the dataset of 1999. In this graphic no co-link analysis was performed to show that early Dutch bloggers are linking outward but do not receive many links back. Due to the lack of reciprocal links the early Dutch blogs do not interconnect into a blogosphere. The Dutch blogs were retrieved from the Internet Archive Wayback Machine between 1999 and 2009 using the Internet Archive Wayback Machine Network Per Year which outputs a network per year. Visualization created by Anne Helmond and Esther Weltevrede with Gephi and in Adobe Illustrator, 2011.

Cluster analysis over time

In 2000 the Dutch blogosphere was dominated by bloggers on personal homepage providers (blue) and student pages (pink) (see figure 8). The left hand side of the map shows a loosely defined news-tech cluster of Dutch news sites, surrounded by U.S. and U.K. news and tech blogs. Similar to the early U.S. blogosphere (Stevenson, 2010), tech and news are prominent in the Dutch blogosphere. The right hand side of the blogosphere shows a cluster of Dutch homepages (~) and student homepages, indicating bloggers manually coding their own blogs before the rise of blog software and hosted blog services. The free homepage provider DDS and Dutch Internet service provider XS4ALL are the most prominent providers. The larger nodes in the center are the founding blogs of the Dutch blogosphere, such as Alt0169, Sikkema, S-lr, Smoel, Rikmulder, Tonie, Prolific, Pjoe, Stronk, Ben Bender, Vandenb, Retecool who form a closely linked cluster. Alt0169.com, a heavy linker in 1999 which didn't receive any links back, has become a central node in 2000.

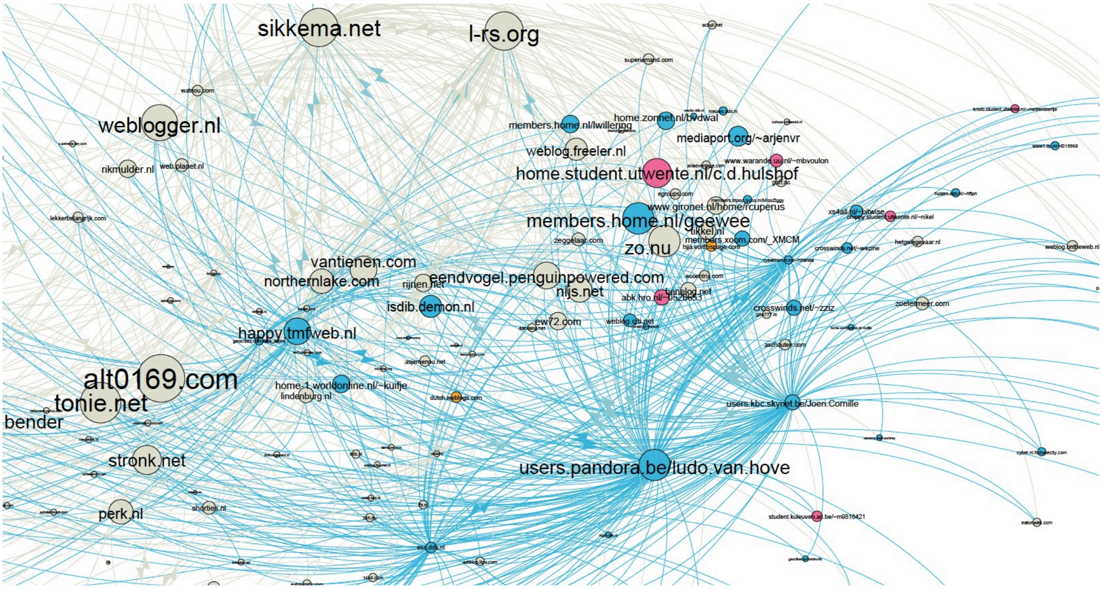


Figure 8: The reconstructed Dutch blogosphere in 2000. All the blogs have been color-coded based on the type of host. Blue shows bloggers on personal homepages, pink shows bloggers on student pages and yellow shows bloggers on early hosted blog services such as weblogs.com and editthispage.com, provided by early blogger Dave Winer. The network graph was retrieved from the G-Atlas, a piece of software developed by the TIC-Migrations group from Paris, to load Gephi networks for further analysis. G-Atlas allows researchers to explore their corpus and can be used as an analytical tool for network statistics and for color coding a corpus. The G-Atlas contains the networks of each historical blogosphere per year between 1999 and 2009 and one composite network of all the years combined. See: https://dutchblogosphere.digitalmethods.net/gatlas/ [Accessed 12 April 2013].

Figure 9 shows the Dutch marketing cluster, which emerged in 2005 and which is still a very dominant cluster in the Dutch blogosphere. Another distinct cluster in the later blogosphere is the Blog.nl cluster, a Dutch network of themed blogs and hosted blog service provider. Blog.nl has a very distinct shape because all Blog.nl blogs list and link to the other blogs on that service as can be seen on the left in figure 11.

Using the same method for coding actors, I created and coded several categories of actors in the blogosphere. First, types of blogs: Homepages, University Homepages and Hosted Blog Services. Second, other actors: Blog-Related Services, Social Media Platforms and Statistics. The categorization was created through expert URL reading and was iteratively complemented with new findings throughout the project. In order to trace specific transitions in the Dutch blogosphere I coded them in Google Refine. I categorized and colored actors belonging to a specific category in Gephi making it easier to locate actors and track changes over time. This method allows for analyzing the role of blog-related services and social media platforms in the blogosphere over time.

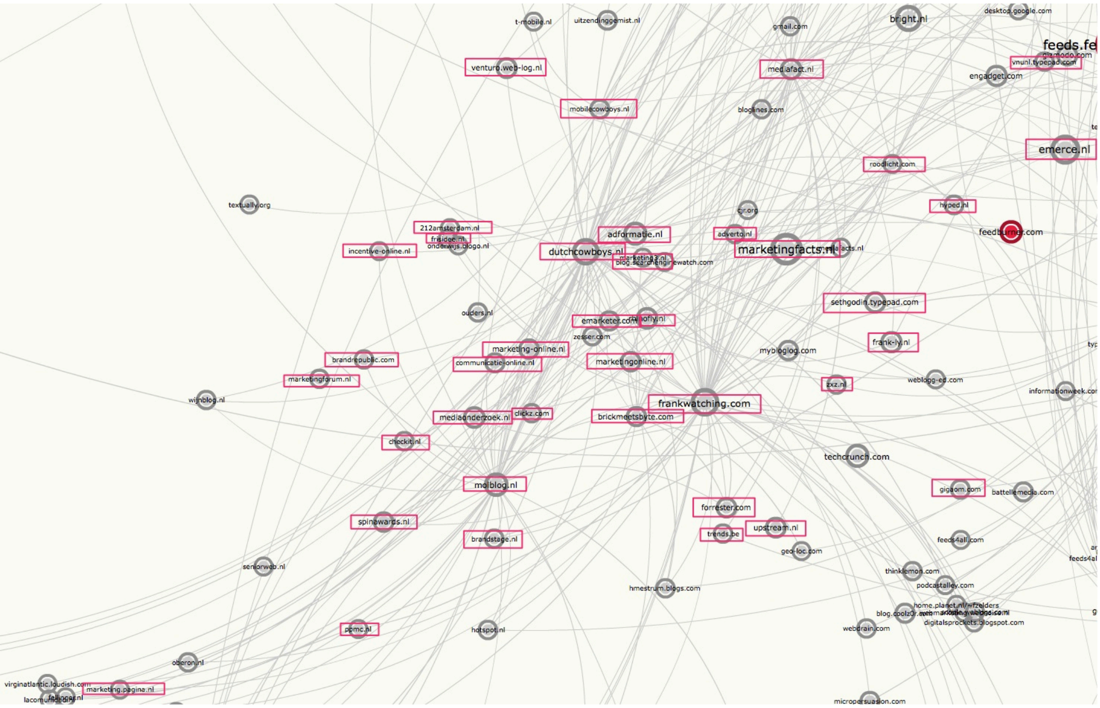


Figure 9: The Dutch marketing cluster in the reconstructed Dutch blogosphere in 2005. The historical blogosphere was constructed from the outlinks of a custom collection of archived Dutch blogs retrieved from the Internet Archive Wayback Machine on which co-link analysis was performed. The network graph was retrieved from the G-Atlas, a piece of software developed by the TIC-Migrations group from Paris, to load Gephi networks for further analysis. G-Atlas allows researchers to explore their corpus and can be used as an analytical tool to color-code a corpus and because it outputs corpus statistics. The G-Atlas contains the networks of each blogosphere from 1999 to 2009 and one composite network of all the years combined. See: https://dutchblogosphere.digitalmethods.net/gatlas/ [Accessed 12 April 2013].

Blog-related software: Statistics

The newly defined blogosphere includes a variety of blog-related actors. As argued previously, the blogosphere does not only take shape by the interconnections between blogs but also by the interconnections between blogs and other actors, such as links to external blog-related services and social media platforms. Blog-related services include portals, manual and automatic blog indexers, external comment services and statistics providers. Aleena Chia similarly argues for “thinking about blogging as a system” which entails moving

> beyond blogs as discrete entities and to understand blog networks as nodes of content production and linkage within an ecology of platforms that include search engines, such as Google.com; blog trackers, such as Technorati.com; social bookmarking websites, such as Delicious.com; link aggregators, such as Digg.com; RSS readers, such as Bloglines.com; and web traffic analyzers, such as Alexa.com (2012, 429).

Figure 13 shows how social media platforms are important actors in this platform ecology, as will be discussed in the next part of this chapter.

One of the most prominent nodes since 1999 has been Nedstat, the Dutch statistics provider. Nedstat—and its basic/free service Nedstatbasic—is a Dutch service providing statistics for web masters and bloggers about their visitors and has been present in the blogosphere together with other statistics providers. Most bloggers publish their statistics, which supports the claim that "the blogosphere is obsessed with measuring, counting, and feeding" (Lovink 2008, 30). Zooming in on the node (see figure 10) shows us all the linked bloggers, presumably using Nedstat as their statistics provider.

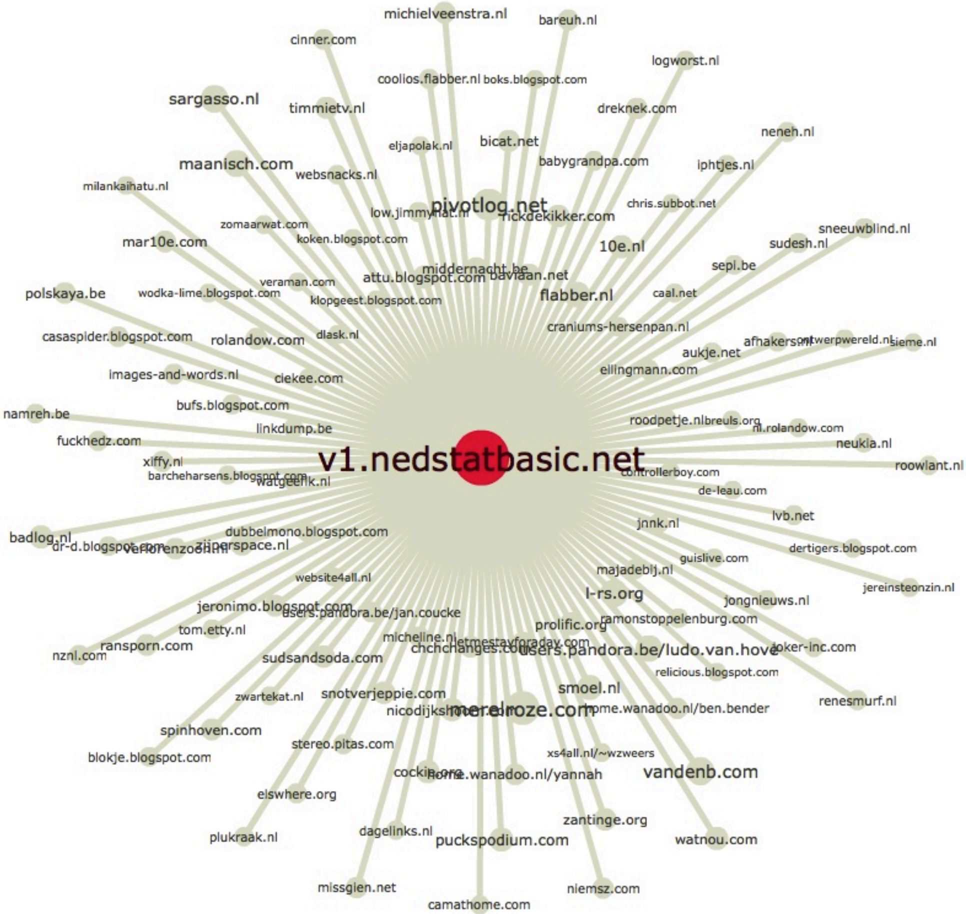


Figure 10: Bloggers linking to Nedstat Basic, a Dutch statistics provider, in the reconstructed Dutch blogosphere in 2004. The historical blogosphere was constructed from the outlinks of a custom collection of archived Dutch blogs retrieved from the Internet Archive Wayback Machine on which co-link analysis was performed. The network graph was retrieved from the G-Atlas, a piece of software developed by the TIC-Migrations group from Paris, to load Gephi networks for further analysis. G-Atlas allows researchers to explore their corpus and can be used as an analytical tool to color-code a corpus and because it outputs corpus statistics. The G-Atlas contains the networks of each blogosphere from 1999 to 2009 and one composite network of all the years combined. See: https://dutchblogosphere.digitalmethods.net/gatlas/ [Accessed 12 April 2013].

Social media analysis: The platformization of the blogosphere

The early blogosphere is characterized by larger nodes such as Alt0169, Sikkema, ~wzweers, the founding fathers of the Dutch blogosphere. The heydays of the Dutch blogosphere are characterized by the rise of specific clusters, such as the marketing cluster (see figure 9), blog-related services such as statistics (see figure 10) and the hosted blog service cluster of Blog.nl (see figure 13). Social media and content links characterize the later period of the Dutch blogosphere from 2004 onwards. In this part of the research I developed methods for analyzing the practices of bloggers and the links between blogs and social media more closely.

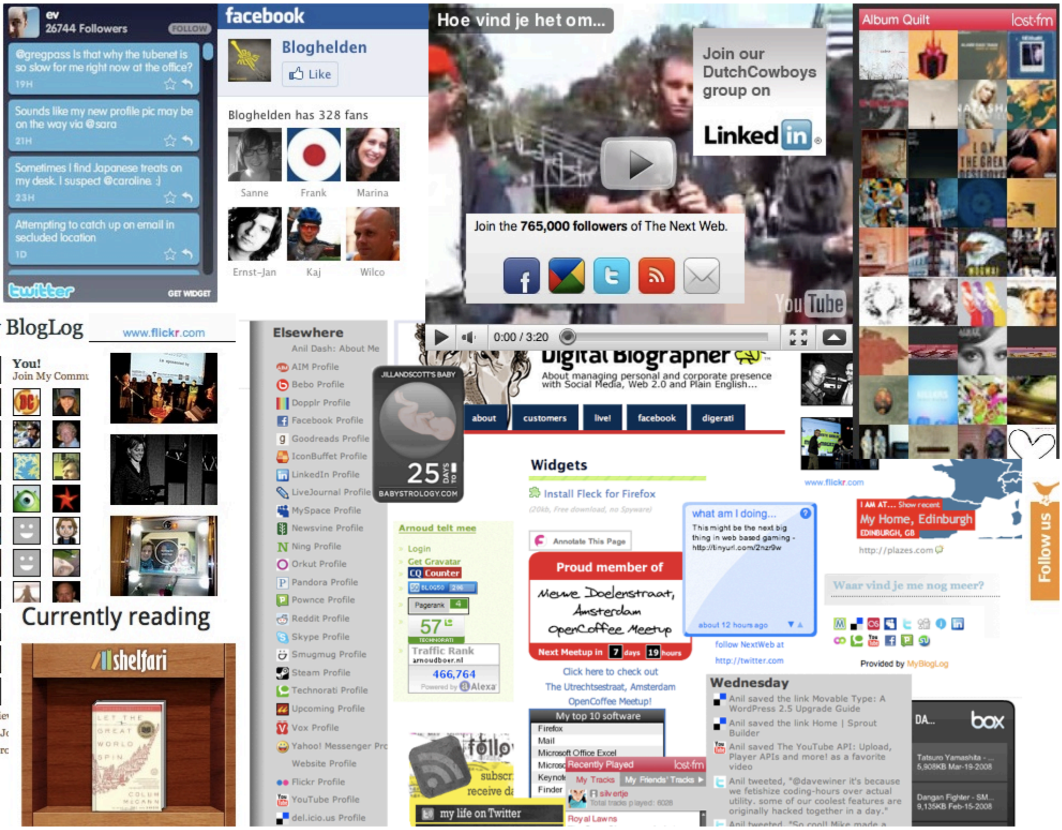
Frank Schaap empirically researched what he calls "the dichotomous nature of the Dutch blogosphere" caused by the clear division between two distinct types of weblog forms: the "linklog" and the "lifelog" (2004). In addition to his categorization, I propose to include the "platformlog" as a third type of blog with particular characteristics. Whereas lifelogs primarily post about daily life in a diary style and in most cases only link to their about page, their off-line contexts and other bloggers, the linklogs link abundantly to other blogs and media in their role of pointing out the best of the web (Schaap 2004). Ignacio Siles describes how these early linklog bloggers were "creating and sharing online navigation sequences through hyperlinks as filtering, curating, or pre-surfing the Web's content" (2011, 744). Later bloggers, Siles argues, started to include other types of content beyond links and diary entries such as political commentary which transformed blogs "from online 'filters' into a 'format' for sharing a variety of content on the Web" (2011, 737). When, in the mid 2000s, bloggers started to incorporate social media content and functionality into their blogs, I argue, they further changed the medium of blogs, the practice of blogging and the structure of the blogosphere.

This new blog type, the platformlog, can then be characterized by embedding and linking content from social media platforms like Flickr, YouTube and Facebook and referring to the author's presence on these platforms in sidebar widgets. It is a place where the "networked self" is performed, which is "communicated across collapsed and multiplied audiences" and which "seeks social opportunities for expression and connection" (Papacharissi 2010, 317) on multiple platforms. The platformlog is often used to present what Nancy Baym refers to as "the widgetized self" (2007), or what I have called elsewhere the distributed self across social media platforms (2010).

Baym introduces the term "the widgetized self" to describe a personalized portal filled with widgets that aggregate a user's social network content and activities from various sources onto a single page (2007).[89] Some of the aspects that Baym conceptualized in her "dream portal" were key to the so-called personalized homepages such as Netvibes and iGoogle that were popular between 2006-2008, with *NewsWeek* proclaiming 2007 to become "year of the

[89] These widgets do not only display content but they also allow for sending new information back to these networks (2007). In this sense, Baym is describing a personal page filled with data pours that establish two-way data flows with social media platforms (see chapter 2).

widget" (Braiker 2006). Personalized homepages allowed users to create their own startpages filled with personalized content using widgets. These widgets, technology blogger Om Malik claims, make the "[w]eb user programmable" and have contributed to the "widgetization of the Web" in which the web is broken up "into small, portable pieces" (2006).[90] The widgetization of content is what I have previously described as the modularization of content and features (see chapter 2) and is a key pre-condition for platforms to extend themselves into the web. Within the blogosphere widgetization is visible in the sidebars of blogs where bloggers insert widgets displaying their content on user-generated content sites such as Flickr and point to their presence on social network sites such as Twitter (see figure 11).



ev 26744 Followers

@gregpass Is that why the tubenet is so slow for me right now at the office? 19H

Sounds like my new profile pic may be on the way via @sara 21H

Sometimes I find Japanese treats on my desk. I suspect @caroline :) 23H

Attempting to catch up on email in secluded location 1D

Bloghelden has 328 fans

Join our DutchCowboys group on LinkedIn

Join the 765,000 followers of The Next Web.

About managing personal and corporate presence with Social Media, Web 2.0 and Plain English...

what am I doing... This might be the next big thing in web based gaming - http://tinyurl.com/2nzr9w about 12 hours ago

Anil saved the link Movable Type: A WordPress 2.5 Upgrade Guide

Anil saved the link Home | Sprout Builder

Anil saved The YouTube API: Upload, Player APIs and more! as a favorite video

Anil tweeted, "@davewiner it's because we fetishize coding-hours over actual utility. some of our coolest features are originally hacked together in a day."

Figure 11: Widgetized blogs: A collection of widgets in the sidebars of blogs. Author's collection. 2010.

Whereas in the mid and late 1990s the self was defined on the personal homepage and later on the blog, nowadays the self is also defined and performed on social networking sites and

[90] At the 2007 Next Web Conference, Tariq Krim, founder and CEO of Netvibes—with it's tagline "(re)mix the web"—saw the rise of "the widgetized web" as "the deportalisation of the web" (2007). This deportalization refers to the end of portals run by big media companies with pre-determined content snippets, and the rise of the personalized homepage, or personalized startpage, where users could aggregate content and functionality from all over the web using widgets.

content platforms. Blog software has popularized the creation of the widgetized self with its easy drag and drop widgets that allow bloggers to easily embed content from other platforms into their blog via the sidebar. Bloggers can add widgets to the sidebars of their blogs in the dashboard of their blog software by adding an embed code to a text widget or by installing a plugin that enables a widget with particular functionality such as inserting a blogger's latest tweets.[91]

The sidebar is no longer only used by bloggers to link to other bloggers, using the blogroll, but also to link to their own presence on other platforms such as Last.fm for music, Flickr for photos and YouTube for videos. As my method collects outlinks from the front pages of blogs and subsequently performs a co-link analysis, I argue that the "widgetized self" in the sidebar on the front page can be captured. The links created in the sidebar of blogs through widgets mainly point to social media platforms that appear as new actors in the blogosphere.

Figure 12 shows the presence of the top 10 social media platforms in the structural historical Dutch blogosphere between 2004 and 2009.[92] While social media platform Last.fm was already founded in 2002, it does not appear in the structural blogosphere until 2004 because the co-analysis method requires two bloggers linking to a node to appear on the map. In 2004, the year of the first Web 2.0 conference organized by the O'Reilly Media in San Francisco, we see the rise of social media in the Dutch blogosphere.

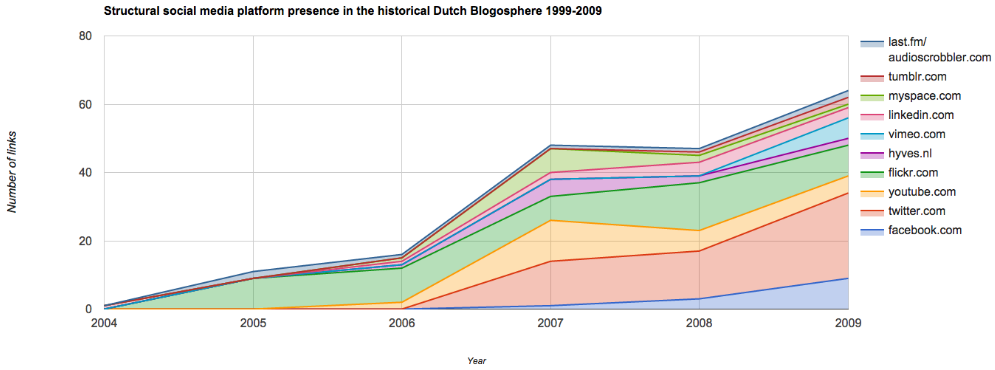


Figure 12: The top 10 social media platforms that were present in the historical Dutch blogosphere between 2004 and 2009. The historical blogosphere was constructed from the outlinks of a custom collection of archived Dutch blogs retrieved from the Internet Archive Wayback Machine on which co-link analysis was performed.

In the next step I developed a method to map the linking practices of Dutch bloggers in relation to social media. In traditional hyperlink analysis these social media nodes are disproportionally large because all references to user profiles on Twitter will collapse into a

[91] See: https://wordpress.org/plugins/tags/sidebar [Accesssed 1 May 2015].

[92] Before 2004 only iMood is present in the dataset. iMood was launched in 1999 as a site for broadcasting your mood on the internet and to friends, see: http://www.imood.com/ [Accesssed 1 May 2015].

single Twitter.com node. I further extended the method described in the section on 'Defining the actors' in which I created custom actor definitions for social media platforms since the actor is often defined after the slash, e.g. twitter.com/username.

Next, I also distinguished between the types of objects that can be linked to in many social media platforms, e.g. bloggers can link to Twitter user pages, status updates, hashtag searches as well as the domain at large. Attuning to the platform-specific types of objects, I created custom actor definitions per platform to enable a more fine-grained hyperlink analysis (see Appendix B). When comparing the 2009 blogosphere with and without this custom actor definition (see figure 13), it becomes apparent that the social media platforms privilege such a fine-grained analysis. Social media are the big nodes in the network without custom actor definition; however, with custom actor definition the social media platforms seem to lose prominence in the blogosphere.

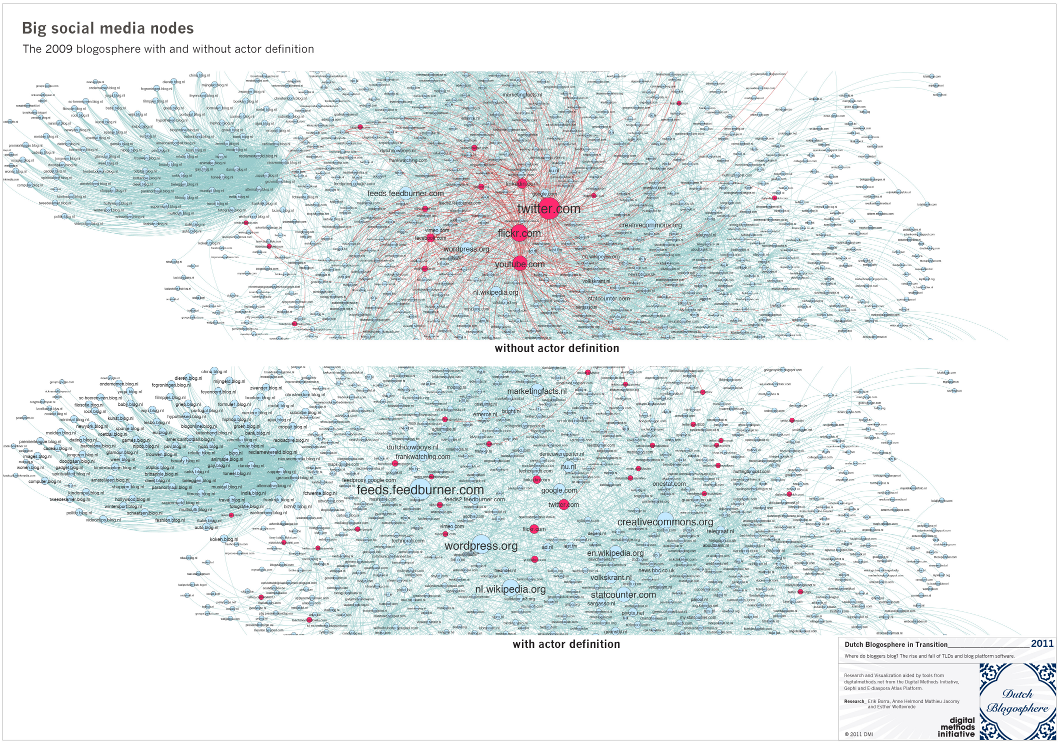

Big social media nodes
The 2009 blogosphere with and without actor definition
twitter.com
flickr.com
youtube.com
feeds.feedburner.com
creativecommons.org
wordpress.org
nl.wikipedia.org
statcounter.com
marketingfacts.nl
without actor definition
marketingfacts.nl
dutchcowboys.nl
feeds.feedburner.com
google.com
twitter.com
creativecommons.org
wordpress.org
en.wikipedia.org
volkskrant.nl
nl.wikipedia.org
statcounter.com
with actor definition
Dutch Blogosphere in Transition 2011
Where do bloggers blog? The rise and fall of TLDs and blog platform software.
Research and Visualization aided by tools from digitalmethods.net from the Digital Methods Initiative, Gephi and E-diaspora Atlas Platform.
Research_ Erik Borra, Anne Helmond Mathieu Jacomy and Esther Weltevrede
© 2011 DMI
digital methods initiative
Dutch Blogosphere

Figure 13: Big social media nodes in the network graph depicting the reconstructed Dutch blogosphere of 2009. The upper graph shows the 2009 blogosphere using traditional hyperlink analysis methods, which collapse all links to social media platforms into a single node. The lower graph depicts the same network after the custom actor definition was applied to distinguish between the various types of links to social media platforms such as links to user profiles and platform content. By distinguishing between different link types it is possible to analyze the role of social media platforms within the Dutch blogosphere in detail. The historical blogosphere was constructed from the outlinks of a custom collection of archived Dutch blogs retrieved from the Internet Archive Wayback Machine on which co-link analysis was performed. Visualization created by Anne Helmond and Esther Weltevrede with Gephi and in Adobe Illustrator, 2011.

The question then arises what do bloggers link to in social media: to user pages or to content (e.g. video, photo, status update)? Figure 14 shows the large social media platform nodes, containing smaller nodes. Comparing the various social media platforms, the results suggest that certain platforms can be defined as 'media sharing' platforms, such as YouTube and Flickr, whose links in blogs mainly consist of embedded content. In the blogosphere map with actor definition, these nodes decrease in size (see figure 13). Facebook is a relatively small node in the Dutch blogosphere and the links it receives dissolve into a divers set of profiles, pages, apps, events, and groups. Hyves—the Dutch social network that outnumbered Facebook in the Netherlands until July 2011[93]—is one of the smallest social media references.

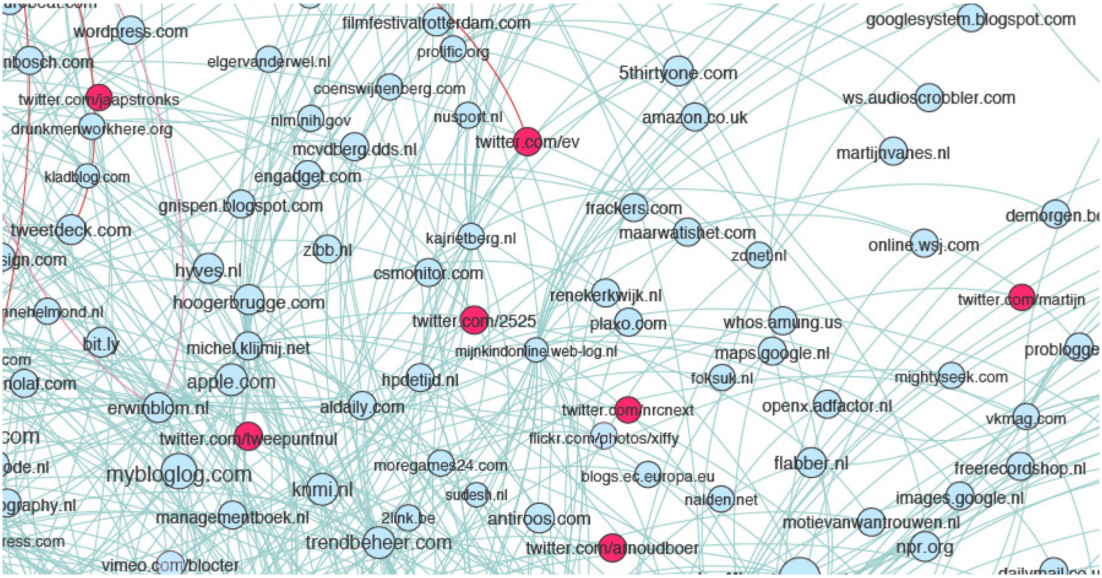


Figure 14: Dutch bloggers linking to Twitter profiles in the network graph depicting the reconstructed Dutch blogosphere of 2009. This graph is based on the custom actor definition to distinguish between different types of links to social media platforms within the Dutch blogosphere. The historical blogosphere was constructed from the outlinks of a custom collection of archived Dutch blogs retrieved from the Internet Archive Wayback Machine on which co-link analysis was performed. Visualization created by Anne Helmond and Esther Weltevrede with Gephi and in Adobe Illustrator, 2011.

[93] At the time of writing, in June 2011, Hyves was still the number one social network in the Netherlands, despite the competition from Facebook and Twitter (comScore 2011). However, Hyves was surpassed by Facebook in July 2011 (Radwanick 2012) and ceased to exist as a social network two years later. On October 31st, 2013 Hyves announced its transformation from a social network into an online gaming portal. Users were offered a mere two weeks to download a copy of their own data before the social network was shut down on December 2nd (Naaijkens 2013). In response to the announcement of the complete shutdown of the oldest and biggest Dutch social network and the loss of a big piece of national digital heritage, the ad-hoc archiving collective Archive Team started a project to archive Hyves. With the help of the distributed archiving power of participants they managed to save all publicly available Hyves data before the shutdown. More information available at: http://archiveteam.org/index.php?title=Hyves [Accessed 18 March 2014]. The Hyves back-ups can be downloaded from: https://archive.org/details/hyves [Accessed 18 March 2014].

Although the Dutch blogosphere prefers Dutch software and services, as I have argued elsewhere (2012), this is not reflected in social media platform links. Twitter, the largest node in the network is a platform that mainly receives links to user pages (see figure 13). This means that bloggers refer to themselves or to friends on the micro-blogging platform. Out of the 160 unique bloggers who link to Twitter user pages, 98 also link to themselves. For Twitter, at least, it may be claimed that the widgetized self can be found in the sidebar, as a new actor in the blogosphere.

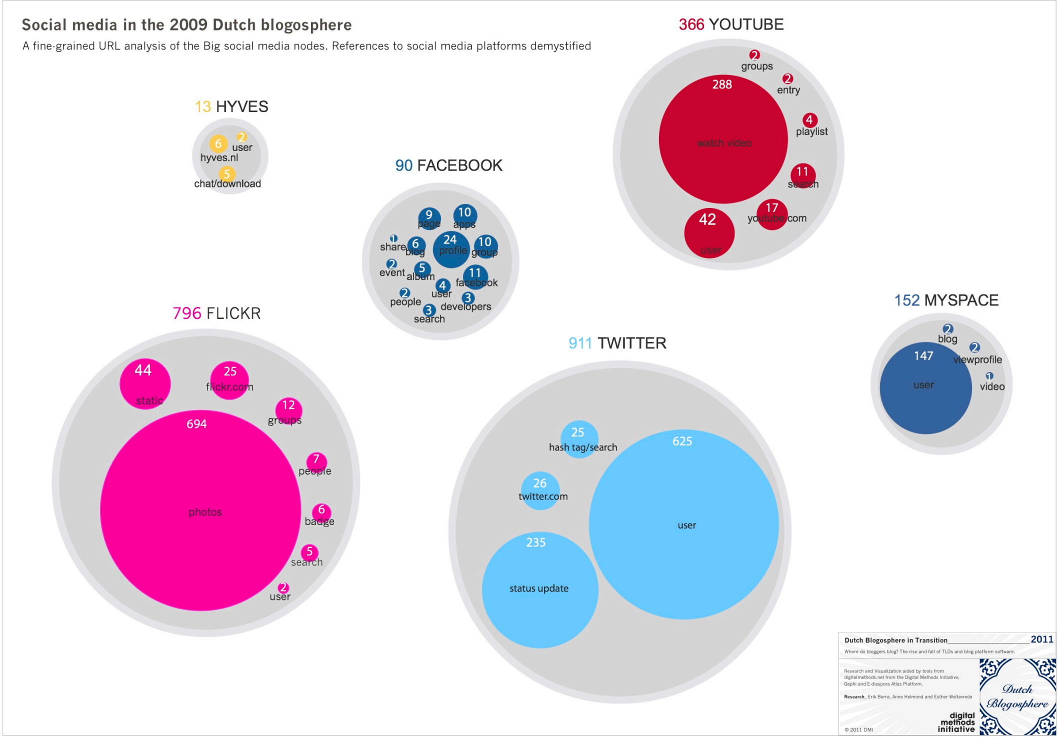


Figure 15: Social media platform presence in the reconstructed Dutch blogosphere of 2009. In this graphic the big social media nodes have been deconstructed based on the custom actor definition and a fine-grained URL analysis to distinguish between different types of links to social media platforms within the Dutch blogosphere. Each grey bubble represents a social media platform and is scaled based on the total number of platform links. Within each bubble there are colored bubbles that represent the different types of platform links—e.g. user pages, status updates, and hashtag searches—and are scaled based on the number links. The historical blogosphere was constructed from the outlinks of a custom collection of archived Dutch blogs retrieved from the Internet Archive Wayback Machine on which co-link analysis was performed. Visualization created by Anne Helmond and Esther Weltevrede in Adobe Illustrator, 2011.

Traditional link analysis has its limitations when analyzing the share of social media platforms in blogosphere networks. Link analysis zooms out to look at platforms as a whole and treats the entire platform domain as the node and in doing so the individual content link and the individual author link disappear. This study has shown that the uniform large platform nodes are deceptive and require a more nuanced exploration.

With bloggers linking to their own presence on social media platforms and by embedding widgets into their sidebars they have altered the link structure of the blogosphere. In particular sidebar widgets automatically create links to platforms, thereby embedding blogs into the social media ecology. In doing so, bloggers are not only connecting other blogs through traditional hyperlinks but they are also connecting social media platforms through social media widgets. The next chapter (chapter 4) explores these new link types and (semi-) automated linking practices that have been created by social media platforms in more detail.

Further platformization: From the sidebar to the blog post

The rise of the widgetized self in the sidebar can be seen as the first phase of the platformization of the Dutch blogosphere. The second phase is characterized by the introduction and integration of social buttons within blog posts for liking, tweeting and sharing a post. Since this study is based on a structural analysis of a blogosphere on front-page level, it is not possible to detect social buttons in blogs since these buttons are usually placed on a sub-page level, the individual blog post. In the following chapters novel methods have been developed to detect social buttons within websites (see chapter 5) and archived websites (see chapter 6) that may be employed to analyze the collection of Dutch blogs. Future research into the platformization of the blogosphere could use these tools and techniques to trace the rise of social buttons within the Dutch blogosphere.

This chapter aimed to contribute to the growing body of literature on blogs and the blogosphere by proposing new methods to empirically investigate transitions in the historical blogosphere over time. Hence a method was developed and described to create a so-called structural blogosphere on the basis of the medium-specific characteristics of the Internet Archive, allowing for the re-construction of a blogosphere on domain level and not on post level. The advantage of this method is that it allows for a structural blogosphere analysis instead of an "issue" or "event" analysis. A structural analysis of bloggers' linking practices was put forward as a way to examine the role of social media within the Dutch blogosphere.

In this chapter I have presented the first known historical mapping of the Dutch blogosphere to analyze changes in the structure of the blogosphere. The results indicate the first signs of a decline of the Dutch blogosphere in 2009, while social media presence in blogs grows. So while Jason Kottke claimed that social media have taken over the core functionality of blogging, the posting of content and sharing of links (2013), the results indicate that this functionality may have been taken over for one type of blog, the traditional "linklog" (Schaap 2004) or "filter log" (Siles 2011). The same could be argued for the decline of the blogosphere (Zickuhr 2010). Regarding the evolution of blogging I have found that the blogosphere has embraced social media for particular practices. That is, the blogosphere has integrated social media platforms and a consequence the blogosphere is shifting from an interconnected network of blogs to being part of the ecosystem of social media.

Future research may further address the changing linking practices of bloggers by further developing the hyperlink analysis presented in this chapter. This could entail looking into different types of hyperlinks and connectivity mechanisms by moving beyond the traditional hyperlinks, the <A HREF> (out)links, that were retrieved from the front pages. Moreover, I wish to distinguish between traditional hyperlinks and the links created by embed codes and social buttons to further enrich the hyperlink analysis to analyze new connectivity mechanisms. This follows a perspective put forward by Greg Elmer and Ganaele Langlois who argue for developing novel methods to explore and map the new networked connectivity mechanisms of Web 2.0 which have moved beyond the traditional hyperlink (2006; 2009; 2013) (see chapter 4, 5 and 6). In the next chapter I further analyze how widgets and social buttons have created new forms of automated linking and the consequences of these new link types. I trace the changing role of the hyperlink within the era of social media and examine how social media platforms have repurposed the hyperlink and have introduced new link types to fit the underlying logic of their platforms.

4. The algorithmization of the hyperlink: Making data platform ready

The hyperlink as a key natively digital object (Rogers 2013, 13) is considered to be the fabric of the web and in this role has the capacity to create relations, constitute networks and organize and rank content. Strands of hyperlink studies have been distinguished that deal with links as objects that form networks, objects that signify a particular type of relationship and the use and usability of links (De Maeyer 2011). What this study aims to contribute to these approaches to studying hyperlinks is an account of the mediating role of software in the production, uptake, processing and circulation of links by looking into the technical (re)configuration of the hyperlink over time in relation to web devices such as search engines and social media platforms.

First, I address how these devices have handled the hyperlink and how they have reconfigured the link to fit their devices. By distinguishing between 'traditional' manually created hyperlinks and hyperlinks configured by software and platforms I focus on increasing automation in the creation of hyperlinks through platform features. Social media platforms have introduced a number of alternative devices to organize relations between users, web objects and content through web activities of sharing, liking, tweeting or digging enabled by social buttons. These devices are understood as pre-configured platform links, which function as a call into the database initiating data connections with the associated platform. Next, special attention is paid to the role of social media platforms in the automatic reconfiguration of the hyperlink into the data-rich format of shortened URLs through a case study on link sharing on Twitter.

Following Tarleton Gillespie's (2010) and Cornelius Puschmann and Jean Burgess' (2013) work on platform politics, the chapter concludes by addressing the politics of data flows of shortened URLs by closely analyzing the role of the Twitter platform architecture and related third-party services in its link-sharing environment. Such an approach to understanding the reconfiguration of the hyperlink examines the mediating capacities of the platform and foregrounds the specificities of the platform itself. By following the trail of a shared hyperlink on this social media platform, one can begin to unfold the processes and actors involved in turning the hyperlink from a navigational object into an analytical device. Further, I use these findings to discuss the implications of the algorithmization of the hyperlink not only for users but also for the web itself.

Through a historical and medium-specific account of the hyperlink that foregrounds its socio-technical relationship with devices, this chapter revisits the political economy of linking (Walker 2002) within the era of social media. I argue that social media platforms reassemble the existing configuration as the hyperlink enters into a process of algorithmization by shifting from a navigational object to a call into a database.

The link as technical artifact

In 'The World-Wide Web' Tim Berners-Lee et al. describe how much of the infrastructure of the web is made out of HTML (Berners-Lee et al. 1994, 78) in which the hyperlink is “the basic hypertext construct” (W3C 2013a). The hyperlink is considered “the basic structural element of the Internet” (Park 2003) and the “essence of the Web” (Foot et al. 2003) and, accordingly, the fabric of the web. In a technical sense the hyperlink is “a technological capability that enables one specific website (or webpage) to link with another” (Park 2003, 49) where it is often noted that “a hyperlink is not only a link but has certain sociological meanings” (Hsu and Park 2011, 364). In this view, links between websites “represent relationships between producers of Web materials” (Foot et al. 2003) and these relationships may indicate a “politics of association” (Rogers and Ben-David 2008, 499). While links establish technical connections between websites they are also considered to organize various types of social and political relations between actors. In her overview of link studies Juliette De Maeyer discusses how hyperlinks are studied as indicators of social phenomena including authority, performance or political affiliation (2013, 739–740). In short, since its inception the hyperlink has been attributed various roles beyond its function as a technical artifact.

This is in sharp contrast with its original design as made explicit by Tim Berners-Lee in his commentary on the web’s architecture: “the intention in the design of the web was that normal links should simply be references, with no implied meaning”[94] and that a link between two pages does not necessarily imply an endorsement (1997). Eszter Hargittai illustrates the hyperlink’s impartiality through its technical design by describing how “technically speaking, all hyperlinks are created equal. They can be easily inserted into any page with the simple code <a href=“http://abc.xy”>text or image</a>” (2008, 87). This means that “a link, by itself, cannot distinguish fame from infamy” (Finkelstein 2008, 117) and that the different attributions and interpretations of hyperlinks are co-constituted by various actors on the web including users, webmasters, blog software, search engines and social media platforms, each employing them for their own distinct purpose. While Berners-Lee envisioned the link as a reference with no implicit or explicit meaning nor as representing an endorsement, devices such as search engines and social media platforms have taken up the hyperlink as an indicator to represent a variety of relations. They have also actively intervened in the ways the relations between hyperlinks, users, search engines and platforms are organized by technically reconfiguring the hyperlink to fit the purpose of the device.

[94] In 1997 Tim Berners-Lee distinguished between two different types of links on the web, normal hyperlinks and embedded hyperlinks. Normal hyperlinks are technically defined by the HTML properties 'A' and 'LINK' while embedded links refer to embedded objects with 'IMG' or 'OBJECT' (Berners-Lee 1997).

This kind of intervention draws attention to the role of software as "the invisible glue that ties it all together" (Manovich 2013, 8) in creating and mediating these linking relations. Engines and devices not only permeate social relations or support sociality but also contribute to structuring relations between human and non-human actors on the web (Beer 2009; Bucher 2012b; Langlois, Elmer, et al. 2009; Niederer and Van Dijck 2010; Ruppert, Law, and Savage 2013). Hyperlinks are the main web-native objects creating this dynamic web of relations and web devices intervene in the assemblage by mediating the hyperlink through software and platform features. In the context of this chapter, this reconfiguration of the hyperlink is traced in relation to the rise of devices on the web, first through the "industrialization" (Turow 2008, 3) of the hyperlink by search engines, second through the automation of the hyperlink by (blog) software and finally through the algorithmization of the hyperlink by social media platforms.

The industrialization of the hyperlink

In the early days of the web, often referred to as Web 1.0, links were mainly created by webmasters and were considered references or connections between website A and website B (Berners-Lee 1997; Park 2003). However, this role changed over time since, "with the initial extension of the Web, hyperlinks took on an increasing role as tools for navigation, transporting attention from place to place" (Halavais 2008, 51). With links as tools for navigation, new actors on the web such as human-edited directories started to collect these links by aggregating them into a single place as a reference for useful websites.

As the size of the web increased, other actors like search engines began to automatically index and publish these links. The automatic output of a list of links was based on matching metatags (Rogers 2006, 41), but as these types of search engines were prone to spam, a new search engine, Google, took a different approach. Google changed the idea of ranking links by also taking the link structure of the web into account. It treated the link as an authority measure, based on the academic citation index, by calculating a ranking for each link based on the weight of sites linking to it (Brin and Page 1998). The PageRank algorithm calculates the relative value of a site and in doing so, Google determined that not all links have equal value.

In order to gain a higher ranking in Google, webmasters aimed to attract links from authoritative sites. This gave rise to the business of search engine optimization with its strategic link building practices and the buying and selling of links on the black link market (Walker 2002, 73). In this way, Google created an economy of links and within what has been termed "link economy" turned the link into the currency of the web (Walker 2002; Rogers 2002) (see chapter 5). Search engines such as Google now regulate the value of links within this economy and have contributed to the "'industrialization' of the link" (Turow 2008, 3) through their automatic indexing, processing and value determination of links.[95]

This link economy of Web 1.0 was based on a relatively open web environment where webmasters and bloggers manually create links between websites and webpages, thereby creating the fabric of the web. The links are traditional one-way links and point from website A to website B where the link is displayed on the former. The link is openly visible and indexable for various search engine parties. While such practices have mainly been based on webmasters manually creating links between websites, both blogs and later social media platforms have advanced more automated forms of linking. The next section briefly addresses this increasing automation of links by software and, as a consequence, what new types of links have been introduced.

The automation of the hyperlink

In the early days of the web, there were no specific rules for designing or naming your link and webmasters would usually manually construct and name their hyperlinks, for example: http://mywebpage.com/aboutme.html or http://www.yahoo.com/news/sports/ and manually link to other websites.

With the increasing popularity of the web and growing number of websites in the mid 90s, systems were developed to automatically create hyperlinks between key topics (Sotomayor 1998). Scripts and the introduction of content management software also introduced automatically generated links such as http://www.nsf.gov/cgi-bin/getpub?nsf9814 and this often led to a decreased readability of links and also contributed to the lost art of reading hyperlinks. It could also involve link breakage when webmasters changed to a different script or content management software. This led to link design recommendations by Berners-Lee at W3C who made the explicit statement that "Cool URIs don't change" (1998).

In the late 90s a new type of website, the blog, with its characteristic reverse-chronological order, put forward a new problem in linking as early blogs displaying the latest blog post on top had no way to refer to a specific blog entry. The blog community, in collaboration with blog software developers addressed this problem with the creation of a new type of link, the permalink. By giving "each blog entry a permanent location at which it could be referenced--a distinct URL" (Blood 2004, 54), the permalink enabled linking to the web-native unit of the blog, the blog post:

> When the Web began, the page was the *de facto* unit of measurement, and content was formatted accordingly. Online we don't need to produce content of a certain length to meet physical page-

[95] The industrialization is described by Turow in relation to "the growth of an entirely new business that measures an advertisement's success by an audience member's click on a commercial link" where the hyperlink is "the product of a complex computer-driven formula" (2008, 3).

> size requirements. And as the Web has matured, we've developed our own native format for writing online, a format that moves beyond the page paradigm: The *weblog*, with its smaller, more concise, unit of measurement; and the *post*, which utilizes the medium to its best advantage by proffering frequent updates and richly hyperlinked text (Hourihan 2002).

The permalink is a specifically formatted hyperlink developed for the medium of blogs to facilitate stable references between blog posts and it became the default output of links for blog posts in all major blog software.

Based on the permalink, blog software developers introduced a new, semi-automatic way of linking blogs with the introduction of the trackback and the pingback, which automatically notify blog B of a link received from blog A.[96] These automatic link notification systems, enabled by the default settings in blog software, created reciprocal links between blogs, making the link openly visible on both blogs. Trackbacks and pingbacks, in other words, are automated linking mechanisms making previously invisible links on the receiving end visible by displaying them underneath the blog post, usually within the comment space.

Blogs also further extended the notion of user-generated linking, previously mainly reserved to webmasters, by enabling users to leave links in the form of comments in the blog's comment space. Instead of webmasters placing links, users—and consequently spammers—could now also create links. With the opening up of the act of linking, the proliferation of links and comment spam links became a serious problem and affected the link economy.

As a response, Google announced a new link attribute (rel="nofollow") on hyperlinks in 2005 so that links with this attribute would no longer pass on value (Cutts and Shellen 2005). The attribute was widely supported by all major search engines and blog software providers and by implementing it into their software "nofollow" became a standard attribute for all links in blog comments. It was a direct intervention of a leading search engine to devalue software-created links such as pingbacks and user-generated links in the comment space.[97]

These examples of changing link types in the blogosphere show the changing configuration of actors in the political economy of links in Web 2.0 where the relations created by the proliferation of user-generated links and software-generated links and the automatic indexing of these links pose a challenge to the engines. In response, Google actively intervened in the production of hyperlinks by introducing a new link attribute to fit the device in order to prevent a (spammy) disruption of the link economy. The next section moves on to explore how social media platforms and their features have contributed to this increasing automation of the production and circulation of hyperlinks.

[96] Pingbacks were developed to address the spam problems of trackbacks by automatically verifying the request and uses a different protocol than trackbacks.

[97] This may be seen in the way Google describes which types of links should receive this new attribute: "We encourage you to use the rel="nofollow" attribute anywhere that users can add links by themselves, including within comments, trackbacks, and referrer lists. Comment areas receive the most attention, but securing every location where someone can add a link is the way to keep spammers at bay" (Cutts and Shellen 2005).

The pre-configuration of the hyperlink: the link as database call

Since the early 2000s the act of creating and sharing links is no longer a manual task belonging to webmasters but is increasingly being performed by software, engines, but also platforms as this section seeks to address. As Ganaele Langlois et al. have previously argued, the rise of Web 2.0 entails a shift in the actors involved in the creation and distribution of hyperlinks:

> Furthermore, whereas the production of hyperlinks in the Web 1.0, HTML-dominated environment was created by human users, hyperlinks in Web 2.0 are increasingly produced by software as tailored recommendations for videos or items of interest, suggested friends, etc. The technocultural articulations that regulate the production and circulation of hyperlinks are thus different in the Web 2.0 environment from the Web 1.0 environment, particularly with regards to the re-articulation of hyperlink protocols within other software and protocological processes (Langlois, McKelvey, et al. 2009).

The tendency towards the previously discussed automation of links has been further fostered with the advent of sharing as "a distributive and a communicative logic" behind participation on Web 2.0 sites and social media platforms (John 2012, 169). Sharing is seen as the "fundamental and constitutive activity of Web 2.0" (John 2012, 167) where Web 2.0 "relies on shared objects—and avenues for circulating said objects" (Elmer and Langlois 2013, 49). The sharing logic of Web 2.0 is for instance mediated by and coded into social buttons (Van Dijck 2013b) which have become typical features of social media platforms to enable the easy circulation of links across different platforms (see chapter 5).[98] Robert Gehl argues that Web 2.0 sites thrive on aggregated and shared content and would appear as empty frames without it: "Without content, these sites are lifeless shells. Without it, Web 2.0 cannot work" (2010, 42).

Many Web 2.0 sites and social media platforms prosper by making use of links and are "as so many Web 2.0 applications seem to be, taking advantage of the openness of the Web and the underlying associations embedded in its link structure" (Nisenholtz 2008, 131). In this context it is of particular interest to look at a new type of link aggregator[99] that became very popular in Web 2.0: the social news website. These sites are of particular interest here because they have introduced new ways of automated linking enabled by social buttons (see also chapter 3 and 5). Digg and Reddit are seen as the prototypical social news sites where users comment and/or vote on submitted stories or links. Users can either submit links directly on the site through a button or on an external website that automatically submits the link to the

[98] While sharing is seen as the main activity of Web 2.0, a large percent of the sharing activity online happens via so-called "dark social." *The Atlantic* technology editor Alexis C. Madrigal uses this term to refer to sharing done via channels which are hard to measure, such as email or instant messaging, which leave no referrer data (Madrigal 2012).

[99] These sites may be seen as precursors of earlier link aggregation sites, or group blogs, such as the collaborative filtering site Metafilter and tech news and discussion site Slashdot.

platform. I detail the development of this automatic submission process within the context of link sharing by focusing on Digg to describe how pre-configured platform links have become embedded in social buttons to enable automated sharing practices between websites and platforms by turning the link into a device to automatically submit and retrieve data.

Initially users could only submit and vote on stories from within the Digg website itself where the site would specifically treat the submission of a story, a link, as a vote. Webmasters and bloggers encouraged users to submit their own stories by directly linking to the Digg stories submit page, but there was no way to submit or Digg a story from an external website. In July 2006, Digg announced a new "code push" to directly integrate Digg into a website (Rose 2006). This new integration functioned in two ways: Firstly, it allowed users to directly "Submit To Digg" from the website they wanted to submit, through the "Digg Story Button"—instead of being redirected to the submissions page—and, secondly, it allowed webmasters to display the number of Diggs for a story, in other words the number of votes for a link. The first step enabled automated link submissions to Digg by "pre-populating the submission form with a title, description, and topic" in the following manner: "http://digg.com/submit?phase=2&url=www.UniqueURL.com&title=StoryTitle&bodytext=StoryDescription&topic=YourSelectedTopic" (Rose 2006). This pre-populated link could be used as the anchor of a Digg image thereby creating a social bookmarking icon that functioned as an external submission mechanism. In a second step, JavaScript could be used to create a 'data pour' (see chapter 2), displaying the number of Diggs for a particular URL and allowing users to Digg or vote on the story directly from the website:

```
<script>
digg_url = 'URLOFSTORY';
</script>
<script src="http://digg.com/api/diggthis.js"></script> (Rose 2006)
```

Combined with a Digg icon this created a social button[100] that, on the one hand, automates link submissions from external websites and, on the other hand, decentralizes Digg features by providing Digg-functionality on external websites. This functionality includes displaying the number of Diggs, or votes, for a particular link and the ability to directly vote on the story if it has already been previously submitted to the site. This process, as enabled by social buttons, can be seen as an important step in the automatic sending and retrieval—or exchange—of data and functionality between websites and platforms, now commonly achieved through APIs, that is Application Programming Interfaces (see chapter 2). This is also explicitly stated

[100] Such icons or buttons are often also referred to as social bookmarking icons. This is derived from their initial function to save or share a page on a platform such as Delicious or Digg. Through a deeper integration with the connected platform the functionality of these buttons has changed over the years by also allowing showing the number of saves or votes or shares or other interactivity related to the page or link. The icons or buttons have since often also been referred to as share buttons or social buttons.

within the Digg documentation, which announced the new submission mechanism as a way for third parties to interact with Digg's data in absence of an API:

> This document outlines the current way we allow partners and third parties to put "Digg This" and "Submit to Digg" links on their site. Our strategy is to bring both types of links to the story submission URL at Digg, which will allow a user to submit the story if it doesn't exist. If it happens to already exist, they will be able to see the story and digg it on our site. The formal API alternative is still in development, and when complete, will allow more direct integration with Digg's data (Digg 2006).

With this new "External Story Submission Process" Digg put forward a new type of linking practice: *pre-configured linking*, where the link is automatically configured for platform submission when clicking the button. The issue here concerns how this practice has not only contributed to the automation of the hyperlink, but also to the algorithmization of the hyperlink by setting up channels for data exchange. This is seen in relation to the increasing modularity of the web (see chapter 2), often associated with Web 2.0 as an architecture of modular elements heavily relying on APIs for the exchange of data (Langlois, McKelvey, et al. 2009) and the rise of social media platforms within this infrastructure. As discussed in chapter 2, the term "platform" could be seen as having a computational, architectural, figurative and political meaning, implying different things for different actors, as put forward by Tarleton Gillespie in his discourse analysis on the politics of platforms (2010). In this chapter I follow the computational meaning of a platform as defined by Marc Andreessen as "a system that can be reprogrammed" (Andreessen in: Bogost and Montfort 2009) to analyze how platforms create an "architecture of assembly" with modular components to disperse and exchange platform data and functionality (Ullrich et al. 2008).

As argued in chapter 2, a social network site becomes a social media platform once it can be programmed, these days typically achieved by providing an API allowing the structured exchange of data and functionality between websites and services. In Web 2.0, or the web as platform (O'Reilly 2005), websites have increasingly become database applications where links in social buttons function as queries into the platform's database and as connection initiators. Technically, the devices that enable the automatic sharing of links are not hyperlinks, rather they are API calls into the platform's database that enable data exchanges, where the hyperlink itself is just one of the many fields within the database.

Due to this particular setup, social buttons allow for a different type of linking than the previously described mechanisms. Social buttons do not create a link between two websites or blogs (see chapter 3), but rather, between websites and platforms, thereby automatically recentralizing all links to the platform (see chapter 5). The platform then aggregates these links and uses them as a voting mechanism in which each link has equal value. This stands in sharp contrast with search engines, where links have different values, even within the same site, and can also be deprived of their value, as in the case of the comment space. However, links shared on platforms do not have equal value in terms of visibility, as some links may only be visible from within the platform itself and to different platform populations.

It may be argued that the social buttons and the associated ranking of links on their platforms have taken up the empty space of the value of the user-generated link and the software-generated link after Google rendered these links worthless with the “nofollow” attribute (Weltevrede 2011). Social media platforms, as actors within the link economy, create their own value from links outside the “nofollow” attribute since while links on platforms have a “nofollow” attribute—which means they do not pass on value for rankings in search engines—they do contribute to the value on the platforms themselves. In this sense, each platform has its own value feature: the Like, the (re)tweet, the +1. As further argued in the next chapter, social media platforms are creating their own web economies related to the link and hit economy by creating platform-specific currencies, often enabled through social buttons.

Social buttons mediate link sharing in particular ways by automating the practice of linking through the pre-configuration of links, while at the same time introducing themselves as intermediaries in link creation and circulation. Platforms emerge as an interface between users, webmasters and search engines and “arise as sites of articulations between a diverse range of processes and actors” (Langlois, McKelvey, et al. 2009). For the political economy of linking in the era of social media, platforms become important actors in the production and distribution of links while at the same time regulating access to these links for engines.

Due to the proliferation of links by the automated linking practices of social buttons and due to the closed character of particular social media platforms, the engines cannot index all links. But, according to Matt Cutts, head of Google’s Webspam team, if Google can crawl Twitter and Facebook links it uses them as a social signal in their web search ranking.[101] While some engines enter direct partnerships with social media platforms,[102] the platforms are establishing themselves as independent new players in the economy of links by building on the industrialization and automation of the link. Besides putting forward more automated forms of linking, social media platforms have repurposed an already existing new type of link, the shortened URL, as an automated device for information harvesting.

Shortened links as data-rich URLs

The idea of URL shortening goes back to 2001 when the service makeashorterlink.com was launched to transform long URLs into shorter ones as an answer to long links breaking—and

[101] On top of that, Google also takes the reputation of the author on Twitter and Facebook into account. See: http://www.youtube.com/watch?v=ofhwPC-5Ub4 [Accessed 24 May 2013].

[102] Google and Twitter had a partnership to include tweets in Google’s Realtime Search results but this deal expired in 2011 (Sullivan 2011). In 2015 they renewed their partnership and tweets are not included in Google on mobile devices (Sullivan 2015). Yahoo! and Twitter previously had a partnership to include tweets in Yahoo!’s search results (M. Mayer 2013) and currently have a deal to enhance a user’s contact cards on Yahoo!’s with the user’s information from Twitter (Dunning 2015). Bing and Twitter also announced a partnership in 2013 (The Bing Team 2013; Ioannides 2013) and closely partners with Facebook (Connell 2013).

becoming unclickable—in email clients.[103] The mainstream success of URL shortening services is often attributed to Twitter, because of the platform's 140-character limit (Antoniades et al. 2011, 715). In a medium where every character counts, users started to employ URL shortening services to save characters by shortening their links. These URL shortening services (USS) do not technically change the link but create an alias for the link, the shortened URL, which then redirects to the original long URL. When the shortened URL is requested the USS issues a "HTTP/1.1 301 Moved Permanently"[104] status code response that tells the browser that the location of the shortened URL has been moved permanently and redirects the user to the location of the original long URL. Shortened URL redirection happens largely unnoticed by users but the technique, which is currently used by more than 600 distinct USS (Maggi et al. 2013, 861), adds an extra layer of indirection to the web by creating an additional hyperlink that needs to be resolved (Schachter 2009).

This new layer of shortened URLs can be analyzed in relation to the algorithmization of the hyperlink by social media platforms as prime contributors to this production of an additional element of infrastructure. It is important to distinguish between two types of shorteners to analyze this new layer of shortened URLs to discuss its implications for various actors on the web. Federico Maggi et al. (2013) differentiate between general URL shortening services such as tinyurl.com, bit.ly and ow.ly and site-specific URL shortening services such as flic.kr (Flickr), youtu.be (YouTube), fb.me (Facebook), t.co (Twitter) and goo.gl (Google).

The general shorteners, popularized by TinyURL, allow you to convert any long URL into a shortened URL after which the shortened URL can be easily shared across various platforms. The second type of shorteners, the site-specific USS, are connected to the website itself and often tied into website features.[105] Social media platforms in particular use site-specific USS to integrate them into their platform-specific sharing features. In these cases, sharing content from a website or social media platform using social buttons, which through their embedded pre-configured linking practices enable the easy sharing of content outside the boundaries of the platform, may automatically produce shortened URLs. This is the case, for example, when you share a video from YouTube through the "Share this video" option on the platform. In this scenario, YouTube automatically produces a short YouTube URL to share: http://youtu.be/Yv73cRpbQaE. Or, if you share an article from The Huffington Post on Twitter using the 'tweet' button a Twitter pop-up appears which contains an automatically

[103] See: http://web.archive.org/web/20010713184016/http://makeashorterlink.com/about.php [Accessed 24 May 2013]. Also, a patent filed in 2000 mentions the idea of using "shorthand URLs" (Megiddo and McCurley 2005).

[104] While there are other types of redirection "HTTP/1.1 301 Moved Permanently" is the most commonly used redirect in USS.

[105] However, not all site-specific USS are run by the websites themselves. Rather, some use a general USS such as bit.ly that offers a custom URL service. Bitly is one of the main providers of custom shortened URLs for websites and they power over 10.000 custom short domains including Foursquare (4sq.com), The New York Times (nyti.ms) and The Huffington Post (huff.to), see: http://blog.bitly.com/post/284009728/announcing-bit-ly-pro [Accessed 24 May 2013]. Since everyone can create a custom short domain powered by bit.ly it has become increasingly difficult to distinguish between general USS and site-specific USS.

produced tweet with the article's title and Huffington Post shortened URL: http://huff.to/xs4lYU. This mechanism is similar to the previously discussed automated link submission from Digg, except that the pre-configured link produced is a shortened URL.

In this sense, sharing as the essential activity of Web 2.0 (John 2012) is mediated by platform-specific objects such as social buttons, which may automatically create platform-specific shortened URLs.[106] Users, websites, platforms and USS together contribute to the additional layer in the web's infrastructure by producing shortened URLs.

The proliferation of links through data-rich shortened URLs

Sharing links may automatically produce shortened URLs and in doing so create an extra alias for the existing link. On top of that, most USS create a *unique* shortened URL for each long URL submitted or shared. This means that if 20 people each share the same article from the Huffington Post to Twitter using the 'tweet' button 20 new unique huff.to URLs and 20 new unique t.co URLs are automatically produced. The shared Huffington Post article no longer has a single URL that points to its location on the web, but now exists under an additional 20 unique huff.to URLs and 20 unique t.co URLs which all point to, or rather redirect to, the original URL. The sharing features and infrastructure of the Twitter platform contribute to a proliferation of links in order to track the further sharing of these links within and outside of its platform by gathering statistics for the shortened URL.

The ability to track analytics for shortened URLs was conceived by a second generation of general USS who added new value to their services by not only shortening the URL but also by providing statistics for the shortened link. They transformed the shortened URL into a data-rich hyperlink suitable for analytical purposes. The hyperlink as a shortened URL is no longer a navigational tool but carries information on the clicks and other (user) data related to it. The biggest general USS, Bitly, which processes about 500 million links per month (Kessler 2014), offers metrics including the number of clicks on the bitl.y link, the date and time the link was clicked, the country the link was clicked from and where the link was shared from, e.g. Twitter, Facebook, Google+, email etc.[107] Besides providing data for each unique shortened URL these general USS also offer aggregated data for all related shortened URLs that have been created for the same long URL: many shortened URLs may be resolved to one single global shortened URL to keep track of a cumulative count of statistics for the original long URL (Chhabra et al. 2011, 95).

This capacity of the shortened URL as an analytical device may provide websites with valuable information about the popularity and spread of their content. Indeed, instead of using

[106] Not all sharing features distribute a site-specific shortened URL and it also depends on whether a link is shared from the web or a mobile platform.

[107] Link statistics can be accessed on the Bit.ly website by adding a '+' at the end of any bit.ly link or through the Bit.ly API, see: http://dev.bitly.com/link_metrics.html [Accessed 24 May 2013].

a general USS for statistics, many websites (and in particular social media platforms) have started to implement their own site-specific USS. This has four main advantages: first, websites can use a custom shortener to create vanity URLs, for example technology blog Ars Technica uses the custom URL arst.ch; second, they are not dependent on an external service to provide them with statistics; third, by not relying on an external service to resolve shortened URL aliases—with the risk that if the USS goes down the shortened links created by the USS will no longer resolve and thus break—the shortened URL will work as long as the website it refers to is still online (Winer 2009); and finally, by routing all links through an internal USS they can be connected to a user profile on the platform. At the end of 2009, two major platforms, Facebook and Google, quietly introduced their own site-specific USS by embedding them into their services (Sterling 2009) to track the spread of their links across the web.

The algorithmization of the hyperlink

So far I have discussed the transformation of the hyperlink from a navigational device into an automated analytical device through the proliferation of a new type of hyperlink, the data-rich shortened URL. I now move on to the role of platform infrastructure to analyze the changing role of the hyperlink within social media, by focusing on Twitter. Attention is drawn to the mediation of links, which entails a reconfiguration of the hyperlink into an algorithmic device to fit the platform.

Initially Twitter did not have its own URL shortener and many users relied on general USS to shorten their links before sharing them onto the platform. In 2008, Twitter integrated TinyURL as its default shortener to automatically shorten all links shared on the platform, but later replaced it with Bitly, which offered metrics and better uptime by comparison (Kirkpatrick 2009). In March 2010, Twitter announced it had been testing its own URL shortener by wrapping shared links in private messages with a twt.tl URL (@delbius 2010). A year later, the company officially rolled out its own internal link shortening service across the whole platform with t.co (Twitter 2011). T.co is Twitter's so-called "link wrapper", a service that "wraps" all the links shared on the platform with its own platform-specific t.co shortened URL.

In other words, Twitter routes all links through its own t.co link service and in the process shortens these links to a 22-character t.co URL and wraps them into a data-rich layer. This means that even already shortened URLs, or URLs shorter than 22-characters, are transformed into a 22-character t.co URL. This process is referred to as link wrapping because while technically Twitter changes the "href" of the original URL into a t.co URL it will usually not display this t.co URL; rather, it keeps the full URL in the link code created with

HTML and displays a shortened version of the original URL in a semantically comprehensible way.[108]

Twitter developer Taylor Singletary explicitly states that t.co is not simply a URL shortener,[109] but a URL wrapper and that Twitter wraps all links for two purposes: first, to protect its users from malicious links to phishing or malware websites and second, to understand how users engage with shared links (Twitter 2011). By rerouting all links through their platform, it can detect spam links by matching them against a list of known malicious links and, subsequently, warn for or even block potentially harmful links. Besides providing a phishing protection mechanism the link wrapping creates a shortened URL, which can be used to gather statistics on the link:

> Twitter may keep track of how you interact with links across our Services, including our email notifications, third-party services, and client applications, by redirecting clicks or through other means. We do this to help improve our Services, to provide more relevant advertising, and to be able to share aggregate click statistics such as how many times a particular link was clicked on (Twitter 2012b).

The hyperlink has become an analytical device in the form of a shortened URL used to gather information about link sharing behavior as each link shared on the platform can be traced and connected back to an individual user profile. Twitter's practice of automatic link wrapping creates a data-rich shortened URL that becomes one of the many fields in the platform's database and may be used for different purposes, including metrics and analytics:

> In addition to a better user experience and increased safety, routing links through this service will eventually contribute to the metrics behind our Promoted Tweets platform and provide an important quality signal for our Resonance algorithm – the way we determine if a Tweet is relevant and interesting to users. We are also looking to provide services that make use of this data, an example would be analytics within our eventual commercial accounts service (Garrett 2010).

By automatically wrapping links in tweets with a t.co URL, Twitter makes this shared data on its platform "algorithm ready" (Gillespie 2014, 168) by reconfiguring the hyperlink to fit the platform. The automatic processing of the hyperlink and its reconfiguration into an analytical device in order to become part of an algorithmic system is what I refer to as the algorithmization of the hyperlink. Hyperlinks in tweets may serve as signals in various algorithms that structure, filter and recommend content on the Twitter platform including the relevance algorithm behind Twitter Search, the popularity algorithm behind Trending Topics and the personalization algorithm behind the Discover tab to find new and relevant content:

[108] "url": http://t.co/9YEaNmtd
"display_url": "huffingtonpost.com/2012/01/14/cos... "
"expanded_url": http://www.huffingtonpost.com/2012/01/14/costa-concordia-disaster-_n_1206167.html.

[109] "'Shortening URLs' isn't a primary focus or 'purpose' of t.co. Wrapping URLs for safe redirection & abuse prevention is the primary goal of t.co, regardless of a URL's original length" (Singletary 2011).

> To generate the stories that are based on your social graph and that we believe are most interesting to you, we first use Cassovary, our graph processing library, to identify your connections and rank them according to how strong and important those connections are to you.
>
> Once we have that network, we use Twitter's flexible search engine to find URLs that have been shared by that circle of people. Those links are converted into stories that we'll display, alongside other stories, in the Discover tab. Before displaying them, a final ranking pass re-ranks stories according to how many people have tweeted about them and how important those people are in relation to you (Twitter 2012a).

David Beer discusses how "the activities of content generation and participation of Web 2.0 feed into relational databases and are then used to sort, filter and discriminate in automated ways and without users knowledge" (Beer 2009, 998) where these activities, including link sharing, are mediated by the platform's infrastructure in anticipation of being fed into an algorithm. Links on Twitter become part of multiple algorithms within the platform but they also become part of multiple databases and algorithms outside of the platform. These include "Twitter Certified Products" partnerships such as SocialFlow using links in its AttentionScore™ algorithm and Topsy's link trending and relevance algorithm. But links on Twitter may have also entered multiple relations with engines, devices and other platforms before being posted onto Twitter. These include sharing practices using social buttons and cross-posting practices where users post a link on one platform which then automatically posts it to other platforms. This may be achieved by connecting platforms in the platform's settings, e.g. "Link Your Facebook Profile to Twitter"[110] to update your Facebook friends and Twitter followers at the same time, or by using external tools such as If This Then That.[111] Links passing through various channels become part of the larger link-sharing environment on the web where each device or platform may reconfigure the link in order to fit their medium by making it "algorithm ready" (Gillespie 2014), a practice I refer to as making data "platform ready".

The practice of link sharing on Twitter, in relation to its larger link-sharing environment on the web, can be analyzed from the perspective of the link itself by looking at the role of the platform architecture, related third-party services and other actors involved in the production, proliferation and circulation of hyperlinks to address the politics of data flows of shortened URLs within social media. The following section of the chapter develops this analysis by means of a specific case study.

[110] See: https://www.facebook.com/twitter/ [Accessed 23 April 2013].

[111] The service If This Then That 'Put the internet to work for you' lets you connect web services and social media platforms with each other using APIs. See: https://ifttt.com [Accessed 23 April 2013].

The social life of a URL shared on Twitter

Here, I follow the path of a single URL to draw attention to the reconfiguration of the hyperlink by software and platforms in the practice of link sharing on Twitter. It takes a medium-specific approach by "following the medium" (Rogers 2013, 25) to see how the platform handles the natively digital object of the hyperlink and it reuses those techniques to analyze the actors within the political economy of linking in social media. I propose a method for following shared links on Twitter in order to trace part of the larger link-sharing environment of the social web and the actors involved in the creation, proliferation and distribution of these links.

To outline this method, I focus on one particular link shared on Twitter, a Huffington Post article on the Costa Concordia Disaster with the following URL: http://www.huffingtonpost.com/2012/01/14/costa-concordia-disaster-_n_1206167.html.[112] Because Twitter creates a unique t.co for every link shared on its platform we first need to detect all the instances of the Huffington Post URL on Twitter. This immediately poses a challenge because Twitter itself does not offer a straightforward way to retrieve all the mentions of the link within the platform. The Twitter Search web interface allows you to search for a URL, but does not return a count for the number of tweets the URL is referenced in nor provide historical data for "old" shared links. The Twitter APIs, in this case the Search API and Streaming API, are the industry-preferred methods to get access to Twitter data, but while the Search API allows you to query for a URL it only provides an index of recent Tweets (6-9 days) which is further limited because "not all Tweets will be indexed or made available via the search interface."[113] The Streaming API limits requests with too many parameters making it unable to track a full URL.[114] Moreover, because it is designed "for receiving tweets and events in real-time,"[115] it does not serve past tweets, making both Twitter APIs unsuitable entry points for the analysis.

Retrieving all mentions for a specific URL can be achieved with the external service Topsy, a certified Twitter partner offering "instant access to realtime and multi-year analyses from the world's largest index of public Tweets."[116] Topsy is a "realtime social search engine" that allows you to search for a topic or URL mentioned on Twitter. Most importantly, in contrast to Twitter itself, Topsy provides historical data making it a suitable entry point for

[112] Initially the data was gathered manually from Topsy on January 16, 2012. It was redone on April 18, 2013 by gathering data from the Topsy API in order to expand the application of the proposed method to historical data.
[113] See: https://dev.twitter.com/docs/using-search [Accessed 25 May 2013].
[114] When using a full URL as the input parameter the API returns an error that the track keyword (the URL) is too long. A canonicalized domain (example.com) is accepted however. See: https://dev.twitter.com/docs/streaming-apis/parameters#track [Accessed 25 May 2013].
[115] See: https://dev.twitter.com/discussions/6971 [Accessed 25 May 2013].
[116] See: https://dev.twitter.com/programs/twitter-certified-products/topsy and http://topsy.com/ [Accessed 25 May 2013].

analysis. The Topsy API allows URL requests to be made and returns a list of unique tweets referencing the requested URL, called Twitter Trackbacks.[117]

The Topsy API returned 172 unique Twitter Trackbacks, tweets containing the specified URL, for the Huffington Post URL.[118] In this case study I am particularly interested in tracing the software mediation from the actors involved in link sharing, so in a next step I extracted all links from the tweets using the Harvester tool which extracts URLs from text.[119] The tool extracted 147 unique t.co URLs from the 172 tweets containing the URL meaning that there are a 147 unique links on Twitter that all refer to the same article on the Huffington Post. These 147 URLs returned by the Topsy API are all t.co links because Twitter automatically transforms all links passing through its platform into t.co links using the t.co link wrapper. The single Huffington Post URL has now been proliferated into 147 unique URLs by the platform.

In the next step, I followed the redirection paths of all 147 unique t.co URLs to trace the origin of the link. By following the URL, or more specifically its redirect chain, we can detect which channels, other than the Twitter platform infrastructure, the link has passed through before ending up on Twitter.

So, in a third step, I followed the redirect chain of all these t.co links to see where they resolved using the custom built URL Follow tool.[120] Since t.co links, and other types of shortened URLs, are redirects this tool employs cURL[121] to follow this redirection path. CURL can follow HTTP redirects of (shortened) URLs by iteratively requesting the server's HTTP location header until the final destination is returned. The output of the header shows the path of redirection (see Appendix C) where the input point is a single unique t.co.

In a fourth step, the redirection paths of all 147 t.co URLs were written down in a spreadsheet. This sheet with redirections was exported as a .csv file and transformed into a Gephi file using the Table2Net tool.[122] In a fifth step, all redirections were coded in Gephi with the name of their URL shortener and UTM link tag before removing these and other tags from the URLs.[123] The connections between the links, the paths of redirects, were then

[117] "On Topsy a trackback is a reference to a URL by an author. A tweet with an URL is a Twitter Trackback. This is slightly more expansive usage of the term compared to its original meaning. The Otter API provides the list of trackbacks for any URL that is in the Topsy index," see: https://code.google.com/p/otterapi/wiki/Glossary [Accessed 25 May 2013].

[118] See: http://otter.topsy.com/stats.json?url=http://www.huffingtonpost.com/2012/01/14/costa-concordia-disaster-_n_1206167.html&apikey=secret] [Accessed 1 June 2014].

[119] See: https://tools.digitalmethods.net/beta/harvestUrls/ [Accessed 1 June 2014].

[120] See: http://labs.polsys.net/tools/urlfollow/ [Accessed 1 June 2014]. The URL Follow tool has been custom built by Bernhard Rieder for this research.

[121] "Curl is a command line tool for transferring data with URL syntax." See: http://curl.haxx.se/ [Accessed 1 June 2014].

[122] See: http://tools.medialab.sciences-po.fr/table2net/ [Accessed 1 June 2014].

[123] UTM (Urchin Tracking Module) are tags that can be added to an URL to track extra information about the link and may include the source of the link (the referrer, eg Google or twitterfeed) and the medium (email, newsletter, Twitter). See: https://support.google.com/analytics/answer/1033863 [Accessed 1 June 2014].

analyzed and visualized using Gephi. Figure 16 shows the t.co network of the single Huffington Post article URL.

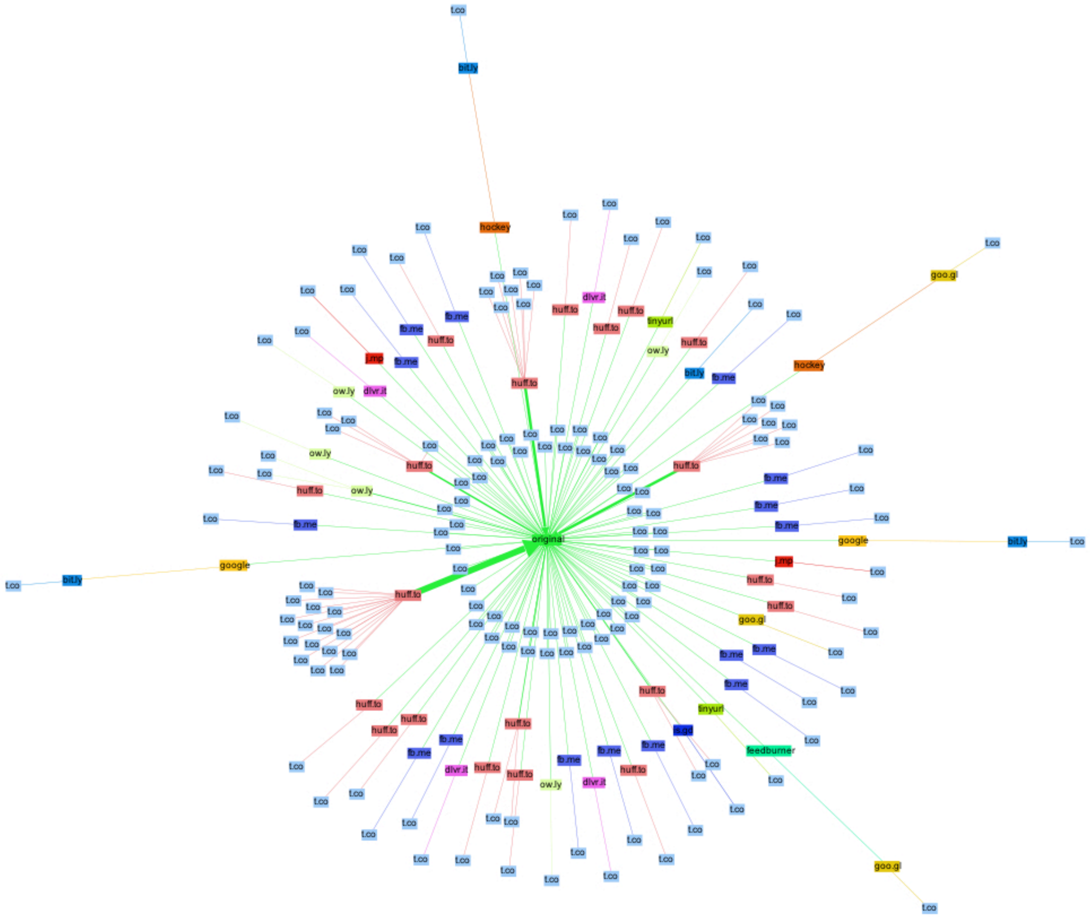


Figure 16: The hyperlink network of a single shared Huffington Post URL on Twitter. The network graph depicts the redirection paths of a single URL. The path resolves from the outside to the inside to the original URL in the middle of the graph. All the redirection paths were coded and color-coded according to their top-level-domain, e.g. t.co and huff.to. Twitter's t.co shortened URLs are colored in light blue. Each node represents a platform or third-party service in the redirection path that has reformatted the original Huffington Post URL into a platform-specific shortened URL to track link data. The data was collected using Topsy, a social analytics company that indexed all public tweets. All the URLs were resolved using the custom-built URL Follow tool to follow URL redirection paths using cURL. The network graph was created and color-coded in Gephi. 2012.

The visualization shows the network of a single link shared on Twitter and depicts the mediating capacities of devices such as engines and social media platforms.[124]

[124] It is a different type of visualization than normally produced as the output of hyperlink network analysis because it does not show the links between different websites but rather the network of redirection paths of single URL.

First, it shows the proliferation of the hyperlink as an effect of its reconfiguration into a data-rich shortened URL. There are 211 nodes (including the original Huffington Post URL) in the graph, which is more than the 147 unique t.co links that have been produced by the Twitter platform architecture. This points us to the existence of other actors involved in the proliferation of the hyperlink that are creating unique URLs for the original URL when handling the link. In this case, I have discovered the following new actors: 20 huff.to (Huffington Post), 16 fb.me (Facebook), 5 ow.ly (Hootsuite), 4 bit.ly (Bitly), 4 dlvr.it (DlvrIt), 3 goo.gl, 2 google, 1 feedburner (Google), 2 tinyurl (TinyURL), 2 j.mp (Bitly), 2 hockey (spam), 1 is.gd (Is.gd).

The network diagram displayed in figure 16 is a visual representation of a directed graph, that is, "every link has a source and a target" (Rieder 2013) and shows the redirect chain of each t.co. In this small sample only 57 of all 147 t.co links, depicted in the center of the visualization, directly resolve to the Huffington Post URL. This means that the other links pass through one or more channels *other* than the Twitter architecture before being posted on the platform. This may happen, for example, if the link is shared from an external website using social buttons, or if the link is shortened using a general USS, or if the link is cross-posted from another platform. I briefly discuss each of these scenarios in relation to the new links found as they point to the various actors involved in mediating distinct sharing practices on Twitter and in creating a new layer of data-rich shortened URLs before moving onto the implications of this new layer.

The huff.to URL is a platform-specific shortened URL belonging to the Huffington Post and is automatically produced through pre-configured linking by using the "tweet" button next to the article. Fb.me is Facebook's platform-specific shortener and is mostly used in the mobile interface indicating that these links have been shared from Facebook's mobile interface to Twitter. By shortening the link Facebook using fb.me can derive value from it by tracing it across the web and feed back these link metrics into its own platform for further use. Ow.ly, bit.ly, tinyurl, j.mp and is.gd are general URL shorteners that can be used before sharing the link on Twitter to gather link statistics but they may also be embedded in third-party Twitter applications as default shorteners. Dlvr.it is a web service to automatically cross-post your new blog content to Twitter, Facebook, Google+ and LinkedIn, and allows you to schedule your posts.[125] Goo.gl is Google's general URL shortener and is also integrated into several Google products including Feedburner for automatically posting your new blog posts to Twitter. Two of the longest redirect chains, where the link passes through three distinct actors, are spam links that have been shortened by several URL shorteners in order to obfuscate the URL by creating long redirect chains (Lee and Kim 2013).

Analyzing the redirect chains of the unique shortened URLs created by the Twitter platform provides an entry point into examining the role of software in the reconfiguration of

[125] To complicate the sharing landscape even further dlvr.it also allows you to use bit.ly as a default shortener instead of dlvr.it making it difficult to establish the actual source.

the hyperlink throughout the link-sharing environment of Twitter. This environment can be seen as a “sociotechnical system” defined by “the intricate collaboration between human users and automated content agents” (Niederer and Van Dijck 2010, 1368) where link sharing is enabled by (semi-) automated devices such as tweet and share buttons, scheduling software, cross-posting services and other third-party applications. Within this system, each actor in the redirect chain reconfigures the hyperlink to fit their platform where many transform it into a data-rich shortened URL to derive value from the associated data flows. In other words, these shortened URLs are created to enable a data connection between the user, the shared link and platform database to track the link and related statistics. The practice of link sharing on social media platforms automatically routes the link through the platform that renders the link, and to a certain extent the user, into a traceable object. As such, this case study unveils a particular strand of actors that could be added to the Twitter data ecosystem described by Puschmann and Burgess (2013): the intermediaries of link sharing, and in particular, URL shorteners and their politics of data flows. These intermediaries derive value from data flows related to the data-rich shortened URL such as detailed link statistics, enabled by cookies.[126] In the case of bit.ly, for example, this “information includes, but is not limited to: (i) the IP address and physical location of the devices accessing the shortened URL; (ii) the referring websites or services; (iii) the time and date of each access; and (iv) information about sharing of the shortened URL on Third Party Services such as Twitter and Facebook.”[127]

Following the web-native object of the hyperlink on a platform has revealed numerous actors that set themselves as intermediaries to track links and associated user data within the link sharing environments of social media. This case study illustrates how the link significantly changed from a navigational device into an analytical device through the automatic reconfiguration by social media platforms. For these platforms, a good hyperlink not only points from A to B, but it is also data-rich by establishing a connection with the underlying platform database and by making data entering the database “platform ready.”

The implications of a web of shortened URLs

Demetris Antoniades et al. describe how “URL shortening has evolved into one of the main practices for the easy dissemination and sharing of URLs” (2011, 715) but they also point out that the shortened URL has hitherto remained an understudied object. They want to put the study of shortened URLs on the agenda since their usage in social networks and social media platforms is rapidly growing; indeed, they account for an increasingly significant amount of web traffic and are becoming a critical part of the web’s infrastructure (Antoniades et al. 2011). According to their work, shortened URLs reflect an “alternative web” which is created

[126] See: https://bitly.com/pages/privacy [Accessed 3 June 2013].

[127] Ibid.

and consumed by a particular community of users on social network sites and social media platforms and this web reflects their interests (2011, 112). In this article, I focused on how users, software and platforms collaboratively and automatically create this alternative web of shortened URLs through the reconfiguration of the hyperlink in the practice of link sharing.

To conclude, I would like to address the implications of this new infrastructural element, which may be seen as part of "the core infrastructure of the social web" (Mattlemay 2011). First, the practice of automatic link shortening adds an extra layer to the web's infrastructure based on centralized hubs (Schachter 2009; Winer 2009). This means that if the USS goes down or the service gets disrupted, then all shortened links from that domain can no longer be resolved to the longer link, which causes link breakage. The Internet Archive and The Archive Team (URLte.am) have both addressed the vulnerability of this new infrastructure with shortened URL archival initiatives. The Internet Archive has a project titled 301works.org, "an independent service for archiving URL mappings," which collects lists of shortened URLs aliases and their corresponding long URL so that the Internet Archive can continue to resolve the shortened URLs in case the participating USS closes down. This initiative requires the active participation and permission of USS which is why the Archive Team, who claim that shortened URLs "pose a serious threat to the internet's integrity," have taken a more aggressive approach by scraping USS to create backups and release those backups as torrent files.[128] Another vulnerability comes from URL shorteners being registered on foreign top-level domains (t.co = Colombia, bit.ly = Libya) which comes with potential legal and censorship issues (Johnson, Arthur, and Halliday 2010; Sandvig 2013).

The extra layer of shortened URLs also slows down the web because redirects impose additional data requests that need to be resolved (WatchMouse 2010). In October 2010 alone, the biggest general USS, Bitly, handled over 8 billion redirects. A year later, it entered a relationship with Verisign, which operates critical parts of the Internet by running two of the 13 root DNS servers (Mattlemay 2011), to ensure fast and reliable shortened URLS (Giles 2011). This partnership also included a data-sharing agreement between Verisign and Bitly, whereby "Verisign's data could add an awareness of activity outside the social sites where Bitly links are used" (Giles 2011). Within this arrangement, moreover, it is pointed out that "an average user interacts more than 30 times a day with Verisign's infrastructure."[129]

A second set of related concerns involves tracking user behavior through shortened URLs and the creation of user profiles by USS by setting cookies (Neumann, Barnickel, and Meyer 2010). This data may be combined with other sources through data-sharing agreements and data may licensed or sold to other parties, depending on the Terms of Service of the USS. To illustrate, in June 2014, Bitly announced a new product called Bitly Audience which will "allow marketers to integrate Bitly data with a CRM software like Salesforce so that if they

[128] See: http://urlte.am/ [Accessed 4 June 2013].

[129] See: http://www.verisigninc.com/en_US/why-verisign/education-resources/how-dns-works/index.xhtml [Accessed 4 June 2013].

want to, say, only send a discount coupon to people who have clicked on two links in addition to signing up for the newsletter or receiving a call from a representative, they can" (Kessler 2014).

Another privacy concern is that not all users are aware that shortened links are publicly accessible and that secret URLs may be found through enumeration or may be indexed by search engines (Neumann, Barnickel, and Meyer 2010). Besides these privacy implications, researchers also point to numerous security implications including the hacking of USS to redirect links to malicious sites or the use of USS in phishing attacks (Chhabra et al. 2011).

Finally, and to return to the changing role of the hyperlink from a navigational device to a database call, one can see that the practice of link sharing as a "fundamental and constitutive activity of Web 2.0" (John 2012, 1) has become mediated by platforms. These platforms are reconfiguring the hyperlink to fit their medium by transforming the link into a data-rich URL in order to make the data platform and "algorithm ready" (Gillespie 2014). In this process, they are automatically creating an additional layer of shortened URLs through the proliferation of hyperlinks by the platform infrastructure. This layer creates additional central hubs on the web, URL shortening services, which not only slow down the web or cause link breakage when the service goes down but may also track users through the practice of link sharing. The mediation of this practice by platforms in the form of link wrapping and URL shortening allows various actors to participate in this new infrastructure and may gain different forms of data, analytics and value for their own platforms.

The next chapter further explores how social media platforms are using platform-specific objects such as social buttons to turn web activities such as liking and sharing into valuable data. In a case study on Facebook's Like button I examine how social plugins function as devices to decentralize platform features and to recentralize valuable user data back to the platform, establishing a data-intensive infrastructure conceptualized as the Like economy.

5. The Like economy: Social buttons and the data-intensive web

Since April 2010, Facebook has increasingly expanded beyond the limits of its platform, offering devices that can potentially turn any website, any app and any web or smartphone user into a part of its platform.[130] A first step towards this expansion was the introduction of the Open Graph in 2010 which allows external websites to link to the platform and create social connections through external Like and Share buttons (Facebook Developers 2012). The possibilities for connecting one's Facebook profile to web objects were further expanded after the f8 Developers Conference in September 2011—an annual event for developers, commercial parties and the public—with the introduction of Facebook actions and objects.[131] Now developers can create apps and buttons that allow users to perform any custom action on any web object. The expansion is driven by the desire to enable more social web engagement, as Facebook CEO Mark Zuckerberg suggests, "making it so all websites can work together to build a more comprehensive map of connections and create better, more social experiences for everyone" (Zuckerberg 2010). In a later interview, he takes the promise of sociality even further: "If you look five years out, every industry is going to be rethought in a social way" (Gelles 2010).

In this chapter Facebook's expansion into the web is examined from a medium-specific perspective, that is, I "follow the medium" and take its ontological distinctiveness (Rogers 2013) seriously by focusing on the role of social buttons and their increasing implementation. This perspective addresses the politics of platforms (Gillespie 2010) and seeks to develop a platform critique that is sensitive to its technical infrastructure whilst giving attention to the social and economic implications of the platform. By tracing the buttons and the data flows they enable, I show how Facebook uses a rhetoric of sociality and connectivity to create an infrastructure in which social interactivity and user affects are instantly turned into valuable consumer data and enter multiple cycles of multiplication and exchange. Building on the previous chapter on the commodification of the hyperlink, Facebook's efforts are linked to a historical perspective on the so-called hit and link economy (Rogers 2002), in which hits and links function as central measurements for user engagement. Doing so, I claim that what is in the making, is not only a social web, but also a recentralized, data-intensive infrastructure which is conceptualized as the "Like economy".

In this Like economy, the social is of particular economic value, as user interactions are instantly transformed into comparable forms of data and presented to other users in a way that

[130] This chapter has been previously published as: Gerlitz, Carolin, and Anne Helmond. 2013. "The Like Economy: Social Buttons and the Data-Intensive Web." *New Media & Society* 15 (8): 1348–65. doi:10.1177/1461444812472322. See acknowledgements on pages iii-iv.
[131] See: https://www.facebook.com/f8/timeline/2011 [Accessed 1 October 2012].

generates more traffic and engagement. Furthermore, the increasing presence of Facebook features on the web contributes to generating connections between websites beyond the traditional hyperlink. The platform advances an alternative form of connectivity which is operating in the back-end and which facilitates participation in Facebook's Like economy by default.

In what follows I first address the emergence of social buttons in relation to specific web economies and introduce the technical specificity of the Like button and the associated Open Graph and Social Plugins. I trace how these features create both data flows between Facebook and external sites and contribute to a reworking of the connections between them, advancing Facebook as one of the central hubs of the web. In an empirical case study I demonstrate the growing presence of Facebook features on the web by contextualizing them in relation to other data-tracking features. Such a perspective draws attention to the changing quality of the fabric of the web, the underlying infrastructure of connectivity between websites. While this fabric of the web has traditionally been studied by tracing mutual linking practices, it is proposed that Facebook's Like economy contributes to the making of an alternative fabric, organized through data flows in the back-end. Next, I address the Like button's capacity to instantly metrify user affects—turning them into numbers on the Like counter—while fostering further user engagement to multiply and scale up user data. Finally, I address the growing resistance to Facebook's creation of a data-intensive infrastructure and conclude by drawing attention to the limits of sociality in the context of the Like economy.

The informational web: The hit and link economy

Since the mid-1990s a number of web-native objects have had a particular stake in organizing economic value production online, most notably the hit and the hyperlink. This section seeks to contextualize the emergence of the Like economy by providing a genealogical account of the different web objects as belonging to and organizing different web periods and web economies.

As previously addressed in chapter 1, the early period of the web is often referred to as Web 1.0 or the "Web-as-information source" and is commonly placed in a dichotomy with Web 2.0 as the "Web-as-participation-platform" (Song 2010, 251–252). Hence, Web 1.0 is addressed as the informational web, an account of the web as a medium for publishing content (Ross 2009). In this context, the number of hits was deployed as one of the first metrics to measure user engagement with a website (D'Alessio 1997). Hit counters (see figure 17) displayed a rough indication of the number of visitors to a page, derived from the number of computerized requests—hits—to retrieve the page, and became the standard for measuring website traffic (D'Alessio 1997). Hits advanced to a central metric for user engagement and thus for web advertising: the more hits a page retrieved, the more attractive it became for placing banner advertisements. The increasing centrality of the hit and its exchange value was conceptualized in the notion of the "hit economy" (Rogers, 2002). While hits cannot be bought or exchanged directly, websites would buy their way into the top of search engines or onto the front page of portal pages in order to attract more hits and so be of more interest to advertisers (Rogers, 2002).

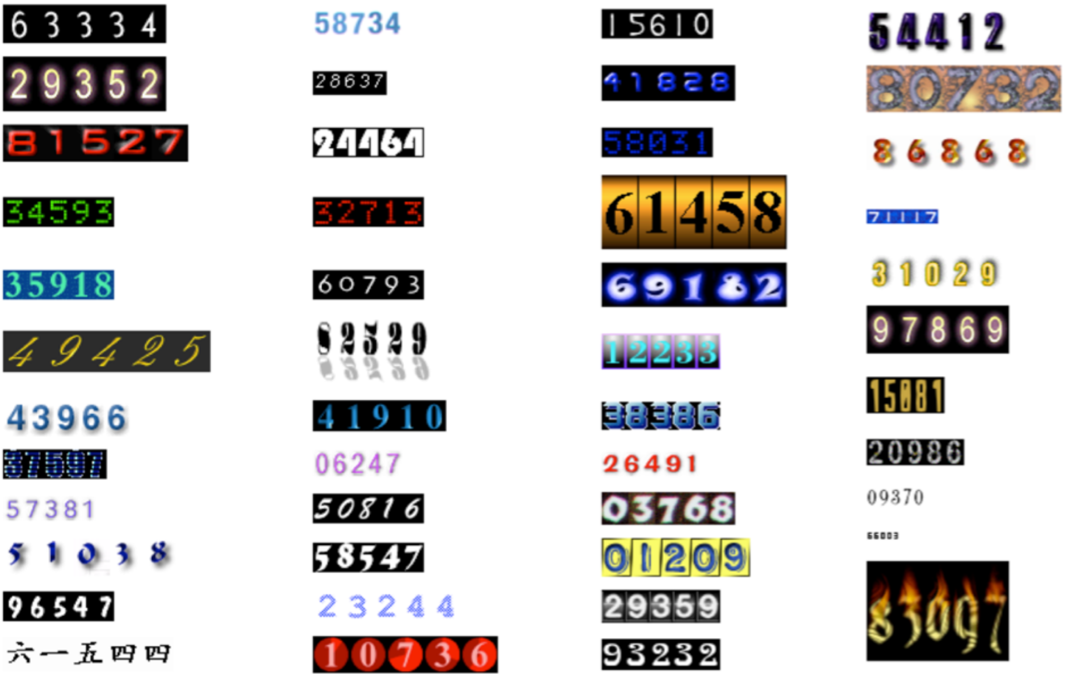


Figure 17: The hit economy—a collection of hit counters that can be placed on websites. The images were collected from Google Image Search using the query ["hit counter"] in April 2011.

The centrality of the hit changed in the late 1990s when a new type of search engine, Google, shifted the value determination of websites from pure hits to hits and links. Inspired by the academic citation index, Google established the role of the link as a recommendation unit on the web by turning it into the main relevance measure for ranking websites (Page et al. 1999). Google founders Brin and Page created the hyperlink analysis algorithm PageRank, which calculates the relative importance and ranking of a page within a larger set of pages, based on the number of inlinks to the page and recursively the value of the pages linking to it. By doing so, the search engine determined that not all links have equal value, as links from authoritative sources or links from sources receiving many inlinks add more weight to the algorithm (Gibson, Kleinberg, and Raghavan 1998).

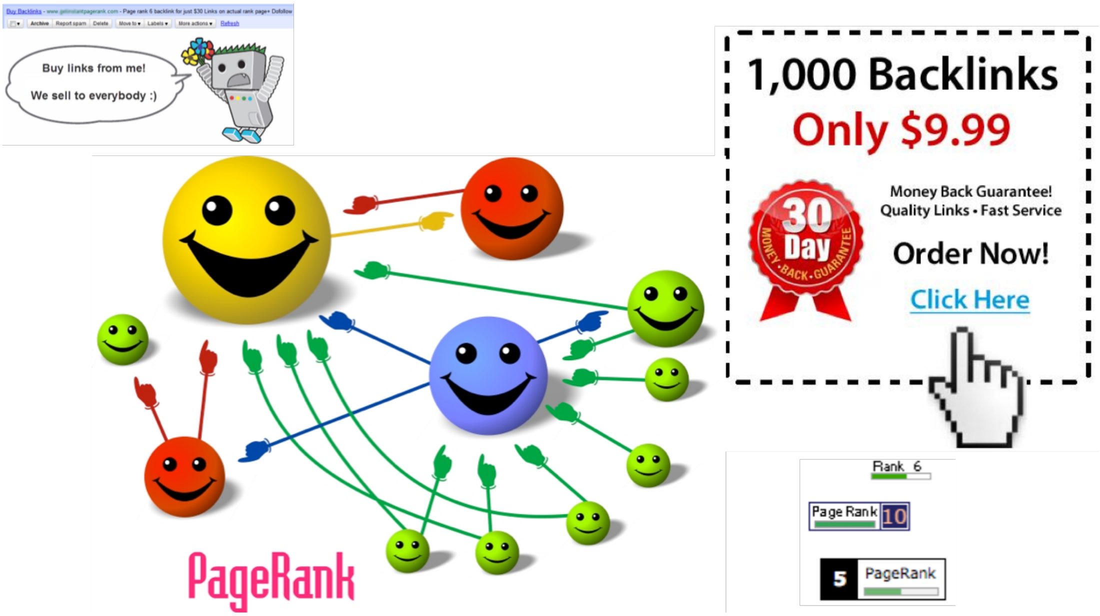


Figure 18: The link economy—advertisements for link markets, a graphic explaining the PageRank algorithm and a number of PageRank buttons. The images were collected from Google Image Search using the queries ["PageRank"], ["PageRank algorithm"], ["PageRank button"] and ["buy links"] in April 2011.

A high PageRank became a quality indicator of a website, and many websites displayed their PageRank with a PageRank button (see figure 18, bottom right). The algorithm established a web economy governed by search engines, not only regulating the value of each site, but also the value of each link this site receives (Walker 2002). Google's increasing centrality has had implications for search engine optimization (SEO) tactics as the focus shifted from optimizing websites to "link-building" techniques—that is, webmasters engaging in mutual linking practices to increase their PageRank. It further gave rise to black markets of links where reciprocal links are traded to improve a site's ranking. These link farms create artificial linking schemes between websites, and are inevitably considered bad linking practices by search engines. But they also contribute to a commodification of links as web objects that can be traded or bought within the "link economy" (Rogers 2002; Walker 2002) (see chapter 4). The move from merely hitting to linking has been a first step towards including relational value in search engine algorithms. However, the social validation largely remains an expert system, since the value of an inlink is determined by the degree of the inlinker's authority.

The social web: The Like economy

The social web proliferated the social validation of web content by gradually allowing for different forms of user participation. While the informational Web 1.0 is characterized by the linking practices of webmasters, the participatory features of Web 2.0 opened up new possibilities for more web users to participate in creating connections between websites (Langlois, McKelvey, et al. 2009) as discussed in the previous chapter. The blogosphere played an important role in advancing the link economy beyond an expert system as, "freed from the 'tyranny of (old media) editors'" (Rogers 2005, 7), blogs offered new possibilities for web users and blog owners to link web content (see chapter 3).

Beyond blogs, it has especially been social media platforms which introduced new features for participation, posing "a group of Internet-based applications that build on the ideological and technological foundations of Web 2.0, and that allow the creation and exchange of user-generated content" (Kaplan and Haenlein 2010, 60). Hence, the social web is defined by the participatory and collaborative production of content, its cross-syndication (Beer 2009), sharing (John 2012) and the relations created between users and multiple web objects—pictures, status updates or pages (Appelquist et al. 2010).

Among the key features to create such connections are social buttons, also referred to as social bookmarking icons, which allow users to share, recommend, like or bookmark content, posts and pages across various social media platforms. As addressed in the previous chapter, these social buttons emerged in the Fall of 2006 in the context of social bookmarking websites like Delicious and content aggregation websites like Digg and Reddit[132] which popularized the acts of sharing and recommending content from across the web by creating buttons that can be placed on any website enabling users to submit or vote for a post on the related platform.

Digg and Reddit were followed by numerous other platforms offering social buttons that afford predefined user activities (e.g. voting, recommending, bookmarking, sharing, tweeting, liking) in relation to the associated platforms, featuring button counters that show the total number of activities performed on the object (see figure 19). These buttons facilitate the cross-syndication of web content and, compared to expert linking practices, introduce a participatory and user-focused approach to recommendation. In addition, the buttons automatically create links between web objects when users click on them through the mechanism of pre-configured platform links embedded in social buttons (see chapter 3).

[132] See: http://web.archive.org/web/20061113041318/http://www.digg.com/tools and http://web.archive.org/web/20061117015427/http://reddit.com/buttons [Accessed 1 October 2012].

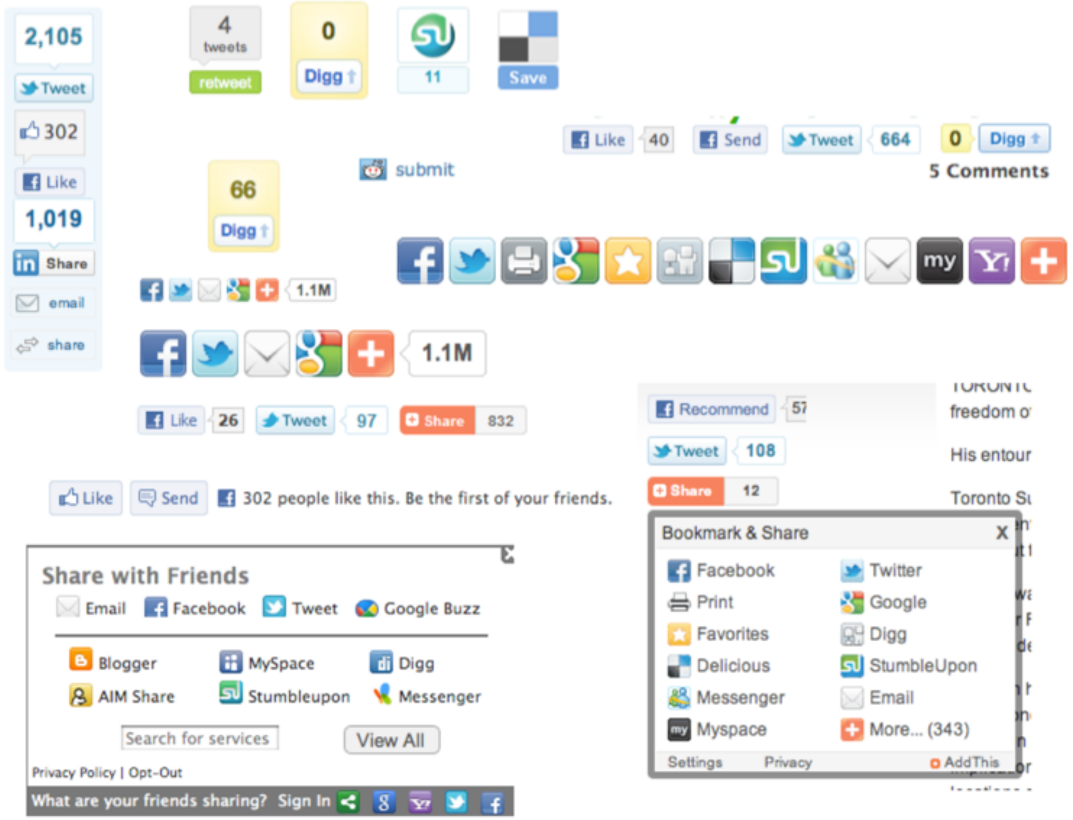


Figure 19: The Like economy—a collection of social buttons on websites. The images were collected from Google Image Search using the query ["social buttons"] in April 2011.

Facebook introduced social buttons with the launch of the share icon in October 2006 as an easy way of sharing web content with one's contacts in order to invoke further social activities on the platform such as resharing, commenting and later liking (Kinsey 2009). The concept of liking had been previously introduced by social network aggregator FriendFeed[133] in October 2007 following user requests for "an ultra-quick way to share their appreciation for the most funny/interesting/useful entries from their friends" (Taylor 2007). Six months before Facebook would acquire FriendFeed (Taylor 2009), liking and the accompanying Like button were introduced on Facebook and presented as a shortcut to commenting in order to replace short affective statements like "Awesome" and "Congrats!" (Pearlman 2009). Liking was put

[133] Social network aggregators are services that aim to aggregate feeds, real-time activity streams, from different social network sites into a single stream (Marshall 2007). Based on the aggregation of RSS/Atom feeds, based on XML (see chapter 2), users can follow their friends' distributed social network activities outside of these networks in one place. Besides following activities, users can often like, share and comment on individual items in the stream. This turns the feed into a distributed activity space, where users can engage with content outside of the social network it originates from. Social network aggregators such as FriendFeed, Socialthing and Plaxo were mainly in use between 2007-2010 (McCarthy 2010).

forward as a social activity that can be performed on most shared objects within Facebook, such as status updates, photos, links or comments. Initially only available within the platform, the Like came with a counter showing the total number of likes as well as the names of friends who clicked it. In 2010, Facebook introduced an external Like button, a plugin that can be implemented by any webmaster, potentially rendering all web content likeable. In 2012, Facebook announced a "mobile Like" that enables app developers to implement a Like action into their own mobile application (Rothbart 2012). This mobile Like Button was introduced at the f8 Developers Conference in 2014 and officially launched for all Android and iOS mobile application developers in October 2014 so that "[p]eople using a mobile app can directly Like the app's Facebook Page, or any Open Graph object within the app, and share on Facebook. The mobile Like Button works seamlessly with the Facebook account the person is logged into on their device, allowing people to Like any piece of content, while in your native app" (Krabach 2014). In doing so, Facebook's Like economy has expanded beyond the web and into the app space (see chapter 7) by enabling data flows between Facebook and external mobile applications using the mobile Like Button which has the ability to turn any app content likeable. Within this chapter however, the main focus lies on the external Like Button for the web to illustrate how Facebook is building a data-intensive infrastructure using its platform features.

According to Facebook, more than 30 million apps and websites are integrated with the platform (Liu 2015), more than 2 billion posts are liked or commented on per day (Facebook Statistics 2011) and there have been over 1.13 trillion likes since its launch in 2009 (Zuckerberg 2012). As of October 2014, Facebook claims that "[o]n average, the Like and Share Buttons are viewed across almost 10 million websites daily" (Krabach 2014). Further, Facebook Like presence is slightly over 30% within the top 10,000 websites (BuildWith 2014).

The external Like button does not only capture actual likes, but also aggregates all activities performed on an object: the number of likes and shares, further likes and comments on stories within Facebook about this object and the number of inbox messages containing this object as an attachment—as the Like is set up as a composite metric.

Facebook's Like button is part of the Social Plugins which allow webmasters to exchange data with the platform and leverage Facebook's social graph. This social graph is a representation of people and their connections to other people as well as objects within the platform and poses a key asset for Facebook. With the launch of the first version of the Open Graph via the Open Graph Protocol at the f8 Developers Conference in 2010,[134] the platform opened up their social graph for external content by providing a way for webmasters to integrate any page outside Facebook into the graph.[135] This integration is mainly facilitated

[134] See: https://www.facebook.com/f8/timeline/2010 [Accessed 1 October 2012].

[135] At the f8 Developers Conference in 2014 Facebook officially opened up the Open Graph for app content with the Like Button for iOS (Sukhar 2014).

through the introduction of Social Plugins, including the external Like button, to enable a personalized, social experience of the web. The plugins allow for a controlled way of exchanging preformatted data between Facebook and the external web as they enable data flows from and to the platform through actions such as liking or by showing which users have engaged with the website or its content within Facebook.[136] These features play an important role in Facebook's strategy of "building a web where the default is social" (Schonfeld 2010) as the Open Graph and Social Plugins mediate the connections between the platform, external websites, apps and users through platform-specific activities. At the same time the Open Graph functions to make external web data platform ready (see chapter 2).

Facebook's data exchange with external sources dates back to 2006 when the Facebook API, an Application Programming Interface that provides a structured exchange of data and functionality between sites and services, was introduced, allowing users to share their data with third-party websites and applications (Morin 2008). Further involvement of third parties was enabled through Facebook Platform in 2007, facilitating external app development within the platform, and in 2008 Facebook Connect was introduced, making it possible to use Facebook profiles as authentication across the web. In 2011, Facebook further reconfigured the integration of external content by expanding the possibilities of app development (Facebook Developers 2012).

The Social Plugins, however, aim at creating an infrastructure in which web users can engage with potentially all web content outside of the platform through Facebook-based activities such as liking, sharing or commenting, setting off a number of data flows and exchange dynamics. Once Facebook users click a like or share button on an external website, this activity is documented on their Facebook Timeline and appears in their contacts' News Feeds and/or tickers[137], while incrementing the Like button counter. The external web content then becomes available for further liking and commenting within the Facebook platform, generating additional data flows back to external counters, once acted upon. More data is flowing from Facebook to webmasters in the form of Facebook Insights providing them with button impressions, which, similar to hits, indicate how many times a Like button was loaded on a page both inside and outside the platform. The Insights tool further features button clicks and anonymized, basic demographic data on likers such as age, gender and location. What is emerging in Facebook's attempt to make the entire web more social is what is conceptualized here as a Like economy: an infrastructure that allows the exchange of data,

[136] The newly introduced mobile Like Button for iOS and Android apps works slightly different from the external Like button for web pages - which is the Like button that this chapter focuses on. The mobile Like Button enables users to like specific app content and shows the number of people who have liked it on a counter next to the button (Facebook Developers 2014b). Unlike the web version of the Like Button, the count remains an aggregate number "x people like this" and does not include the possibility to show the faces and names of friends who have liked this.

[137] The Facebook Ticker "shows you the things you can already see on Facebook, but in real time" in the upper left corner of the News Feed (Facebook Help Center 2015).

traffic, affects, connections, and of course money, mediated through Social Plugins and most notably the Like button.

Whereas the link economy is organized around the *web-native* object of the link, Facebook's Like economy is organized around the *platform-native* object of the Like. That is, social media platforms have created their own platform-specific currencies such as the like, the share and the retweet which have been derived from platform activities such as liking, sharing and retweeting. In doing so, social media platforms have created and introduced new web currencies that are tied to the mechanics and logics of their own platform infrastructure.

Interestingly, not all contributors and contributions to this emerging Like economy are visible or require active engagement with plugins. According to Arnold Roosendaal (2010), the Like button can be used to read a cookie from a user's device, which is issued after creating a Facebook account or visiting any website with Facebook features. From that moment on, the button is tracing the visitor's browsing behavior and is automatically generating data for Facebook by connecting it to individual Facebook profiles. Being tracked by Facebook through such cookies can only be prevented by disabling the use of cookies in the browser options or by installing a browser add-on such as Ghostery that disallows third-party tracking, as discussed in more detail later in this chapter.[138] Most crucially, this does not only apply to Facebook users, the Like button cookie can also trace non-users and add the information as anonymous data to the Facebook database.[139] According to Facebook, this data is used to improve its services but it is also for personalized advertising. Therewith the Like button turns any web user into a potential Facebook user, as each user may unknowingly contribute to the production of valuable browsing data for the platform.

At the f8 Developers Conference 2011, Facebook expanded the possibilities of instant and invisible participation even further, most notably through the aforementioned Facebook custom actions. When creating an app, developers are prompted to define verbs that are shown as user actions and to specify the object on which these actions can be performed. Instead of being confined to 'like' external web content, users can now 'read', 'watch', 'discuss' or perform other actions (see chapter 2).[140] These new apps come with the controversial feature of frictionless sharing and automatically post performed activities to the ticker once users have signed up for an app (MacManus 2011). Whereas the Like button requires an active click to share content, the new actions enable automatic sharing of content or activities. Also, while recommendations via the external Like button direct users to websites outside of

[138] See: https://www.ghostery.com [Accessed 1 October 2012].

[139] Belgian researchers, at the request of the Belgian Privacy Commission, analyzed Facebook's new terms of service and privacy policy and the behavior of social plugins. Their research confirmed that in the Spring of 2015 Facebook still used its social plugins to track users as well as non-users (Acar et al. 2015).

[140] In July 2014 Facebook announced that it has started testing a 'Buy' button for ads and Page posts which enables users to buy an advertised or promoted product from within Facebook, see: https://www.facebook.com/business/news/Discover-and-Buy-Products-on-Facebook-Test [Accessed 28 July 2014].

Facebook, the new actions refer to Facebook-internal app content only, fostering engagement with external content within the platform.

As a consequence, the Open Graph and the external Like buttons create a data-intensive infrastructure enabled by the involvement of a series of actors such as users, webmasters and developers. Webmasters are granting Facebook real estate on their web pages, creating a data pour by embedding the Like button, in exchange for user engagement, platform traffic and user data through Facebook Insights. Users are allowing the use of their data and affects to enable social interaction with other users and to perform their online identity. But third-party actors also increasingly participate in the Like economy, by establishing what is commonly referred to as Like-walls or Like-gates,[141] trading access to content for a click on the Like button of a Facebook Fan Page or by buying likes from external like resellers to increase their fan count—turning likes into vanity metrics—and make their pages more attractive. Similar to the link farms of the link economy, users can buy likes from services that are set up as "like farms" to produce likes on demand (De Cristofaro et al. 2014).

In establishing relationships with webmasters and developers, Facebook is opening its platform in a controlled way, letting carefully selected user data flow outside of the platform in order to maximize data flows into the platform.

Reworking the fabric of the web

I now move on to look into the specific ecology of the Like economy to explore how the multiple processes of exchange are enabled and how Facebook reworks its relation to the web. As discussed in chapter 2, Facebook has been criticized as a walled garden (Berners-Lee 2010), a closed infrastructure, which controls connectivity to data after it has been integrated into the social graph. Objects from within the platform can be shared and linked to from the outside, yet actual access to these objects by following the hyperlink is managed on two levels: First, on the level of access to the platform, defining if login is required to view the object, and second, on the level of privacy settings, determining whether a user has the corresponding access rights to view the object. While access from the outside is carefully regulated, it has been shown that the platform is constantly proliferating possibilities to integrate external

[141] The term Like-wall refers to a Facebook-specific type of paywall, a website monetization system whereby access to the content of a website is restricted unless a user pays for it. In the case of a Like-wall, the currency for gaining access to content is a Like. This practice is also referred to as like-gating content. As of November 5, 2014 when the new Platform Policy for a new version of the Graph API v2.1 went into effect, these actions have been restricted by Facebook by no longer allowing incentives for liking: "You must not incentivize people to use social plugins or to like a Page. This includes offering rewards, or gating apps or app content based on whether or not a person has liked a Page" (Singh 2014). On November 11, 2014 the updated Platform Policy rules state in regard to the use of Social Plugins: "Don't participate in any "like" or "share" exchange programs" (Facebook Developers 2014d).

content and data into the platform, facilitating a decentralization of data production (see chapter 2).

With the introduction of Social Plugins and the Open Graph, Facebook activities such as liking, commenting and sharing are no longer confined to the platform but are distributed across the web and enable users to connect a wider range of web content to their profiles. Social Plugins may also have a decentralizing impact on external websites. Engagement with web content is not confined to designated comment spaces, but takes place across a wide range of platforms and within Facebook across many profiles and News Feeds.

In this context, external websites cannot be considered as discrete entities, but function as initializers for a series of platform-based interactions, as further demonstrated in chapter 6. The more Social Plugins a website integrates, the more it opens itself up to being shaped by the activities of Facebook users. Users also experience such websites in a personalized way, as Social Plugins provide recommendations based on the activities of a user's contacts and feature the engagement of friends with the website. Whereas these are rather novel perspectives for the web, they are key characteristics of social media platforms, which have little original content and are shaped by cross-syndication practices and aggregated content (boyd 2010; Gehl 2014). As a consequence, Facebook and the external web are becoming increasingly interconnected with each other, as the activities performed in one space affects the other, rendering both more open and relational.[142]

On top of that, the Like economy contributes towards a decentralization of actors involved in value creation, as it is reliant on webmasters as infrastructure providers implementing Social Plugins and is dependent on users to engage with Like buttons and liked content. It is the partial opening of the walled garden that poses an incentive for webmasters to participate in the Like economy, since social buttons provide a new way to foster user engagement and traffic. As opposed to the hit and link economy, website traffic is no longer mainly driven by portals, search engines or referrals from other sites. In the context of the social web, traffic increasingly comes from social media platforms, facilitated through the decentralized presence of platform features across the web, where content is shared and has the potential of being reshared to ever more contacts.

But Facebook's efforts to make each and every web experience more social, that is connecting all web experience to its platform, indicates a simultaneous rewiring of the web. Social buttons open up sharing possibilities, yet the connections created by users instantly direct back to the platform as opposed to the reciprocal linking practices of webmasters. While the Open Graph presents an attempt to decentralize opportunities to connect external web content to Facebook, it at the same time recentralizes these connections and the processing of user data.

[142] This relationality also has particular effects on web archiving and using web archives for research, as discussed in chapter 6.

Numerous actors are contributing to the creation of the infrastructure of the Like economy, but not all are given full access to the data they produce themselves. In the case of external Likes, data flows are first of all directed to Facebook and are then fed back in a highly controlled way to other actors involved. Users cannot systematically access their own likes, which are turned into ephemeral objects in the News Feed and on their Timeline. While the successor of the Facebook profile wall, the Timeline,[143] introduced the clustering of activities in relation to different topics and temporal intervals, this only applies to liked Facebook Pages and not to other liked objects. As a consequence, users cannot directly search and use their Likes as a bookmarking system, as external Likes retain their status as fleeting objects for spontaneous engagement. Webmasters who implement social buttons to facilitate wider engagement with their content cannot see how their content is being discussed inside the platform as they are only provided with the aggregate numbers on associated counters and Facebook Insights. The recent introduction of Open Graph actions and frictionless sharing add another quality to the dynamics of recentralization. The new app development features integrate external content even more strongly into the platform, as engagement with the web and mobile services is now promoted via apps rather than external buttons which refer users to content within the platform as opposed to linking to external websites.

In order to extend its data mining and become the central hub of social linking, Facebook is reversely dependent on the dynamics of decentralization as discussed above. Simply because the platform can expand some of its key features into the entire web and integrate ever more objects into the social graph, it can recentralize and monetize the created connections and data flows, as they all direct back to Facebook. The dynamics of de- and re-centralization are not only interconnected, they form a prerequisite for the Like economy. They enable Facebook to maximize its data mining activities while at the same time keeping control over the key entities of exchange—data, connections, traffic and, as shown in the next section, user affects.

But first I exemplify the interplay between de- and re-centralization and discuss Facebook's relevance in relation to other data mining services on the web. For this purpose, colleagues and I have developed a tool—the Tracker Tracker tool[144]—that can detect the presence of third-party trackers on websites.[145] It is built on top of Ghostery, a privacy browser plugin, which recognizes the fingerprints of a number of data mining services active on websites. The plugin allows web users to see the back-end data flows that are initiated when loading a page and offers the option to block them. This research approach repurposes the analytical capacities of the privacy awareness plugin (Rogers 2013), which makes the invisible web visible with its trackers, beacons and cookies. Taking the top 1000 websites according to Alexa as a starting point allows to explore how the part of the web that receives most of the

[143] Timeline is the successor of the Facebook user's profile page and organizes content and activities in a timeline, see: https://blog.facebook.com/blog.php?post=10150289612087131 [Accessed 1 October 2012].
[144] See: https://tools.digitalmethods.net/beta/trackerTracker/ [Accessed 24 March 2014].
[145] For a more detailed explanation and technical description of the tool, see chapter 6.

traffic comes with an embedded data mining infrastructure based on tracking devices. Figure 20 shows the presence of Facebook Social Plugins and Facebook Connect.

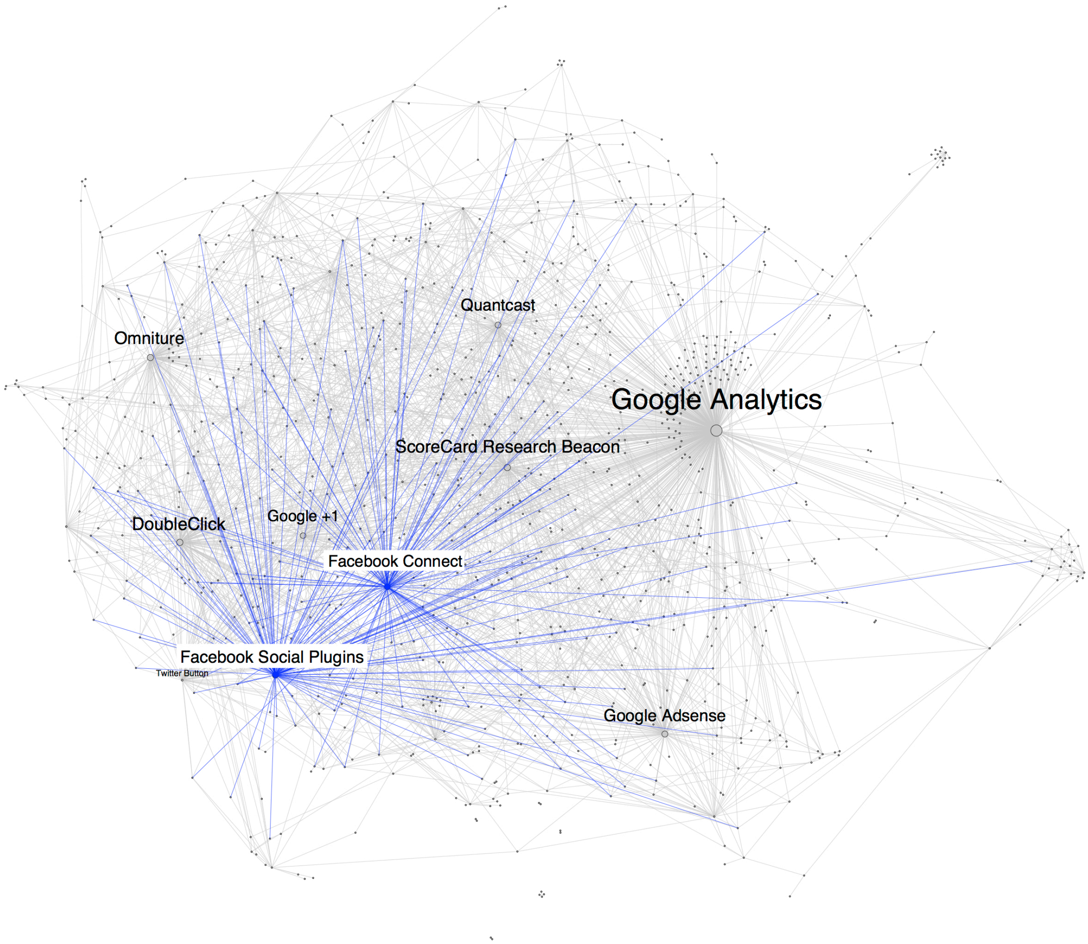


Figure 20: This bipartite graph depicts the top 1000 global websites according to Alexa, February 2012 and their trackers. Websites using Facebook Social Plugins and Facebook Connect are highlighted in blue. The network graph was created with the Tracker Tracker tool, spatialized and color-coded in Gephi and annotated in Adobe Illustrator. 2012.

In this sample, around 18% of all websites feature at least one of these connections to Facebook, allowing users to engage with their content via Facebook features and enabling multiple data flows in the back-end. The second map (see figure 21) shows the overall presence of different types of tracking devices, that is web analytics, widgets (including Facebook's Social Plugins), advertising services and trackers, and allows to draw more general conclusions about the organization of value and the fabric of the web, that is the organization of connections between websites.

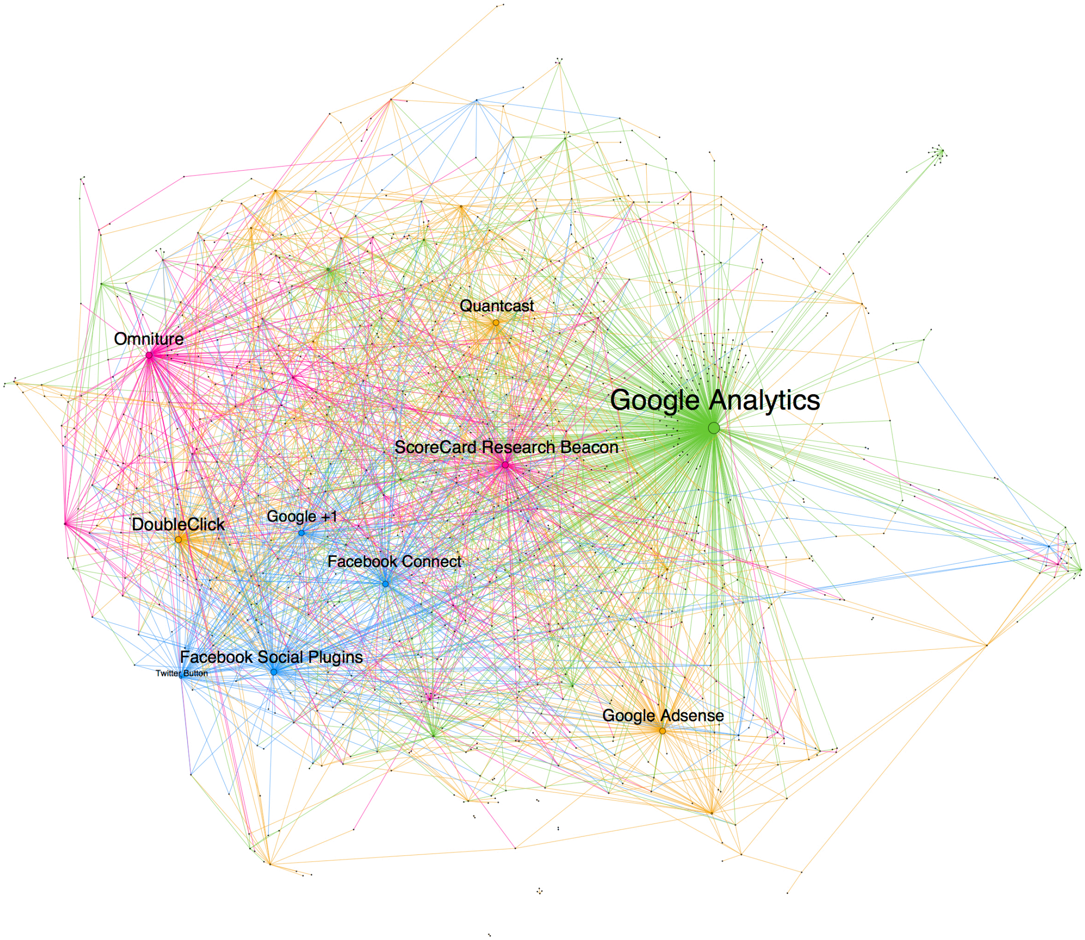


Figure 21: Websites and their trackers in the top 1000 global websites according to Alexa, February 2012. The trackers have been color-coded according to tracker type. The trackers have been color coded according to the tracker types provided by Ghostery.[146] The network graph was created with the Tracker Tracker tool, spatialized and color-coded in Gephi and annotated in Adobe Illustrator. 2012.

Focusing on the presence of tracking devices allows to explore an alternative network of connections, one that is not established through mutual linking practices between websites, but based on associated trackers. According to Ghostery, over 1000 companies are issuing different tracking devices on the web.[147] Despite Google's predominant position, Facebook has established itself as one of the main agents. In its attempt to render web experiences more social, as these maps show, Facebook is fostering an infrastructure of decentralized data

[146] See: http://www.knowyourelements.com/#tab=intro [accessed 12 January 2013].
[147] See: https://www.ghostery.com/about [Accessed 1 October 2012]. In August 2014, this number has since risen to almost 2000.

production and recentralized data processing. Its investment in creating a more social web is hence tied up with an involvement in web economies focused on data mining and web analytics. Such web economies based on cookies have existed since the informational web with its hit and link economy. What is different in the Like economy is that this data collection in the back-end can be connected to the platform's social graph, merging two sources of user data together that had not previously been connected.

Facebook's expansion into the web does not only come with implications for an understanding of the social web and its economic affordances, but also contributes to a new perspective on the fabric of the web. What emerges, when exploring the connections between websites forming through the presence of tracking devices, are clusters organized around major data analytics corporations like Google Analytics, Quantcast, Omniture and, most recently, Facebook. These form an alternative fabric of connections between websites, which are operating in the back-end and are enabled by a range of actors, including webmasters and web users in the front-end. Tracking devices thus establish new relationship markers between websites beyond the hyperlink, as this alternative fabric is not organized through connections between websites, but through the presence of third-party tracking devices on websites linking to associated data mining services (see also chapter 6). The actual connections are enabled through user activities that set data flows in motion from the websites to the associated service. In this sense, the emerging fabric can be understood as live, that is responsive in real time. Yet, as the next section shows, it is also lively (Marres and Weltevrede 2013), as it is changing its intensity depending on user activities and circulation of data.

A web economy of metrification

So far it has been shown that Facebook's Social Plugins do not only enable a social web, but also partake in multiple dynamics of data mining and circulation. I now move on to explore how user engagement is instantly transfigured into comparable metrics and at the same time multiplied on several levels.

A click on the Like button transforms users' affective, positive, spontaneous responses to web content into connections between users and web objects and quanta of numbers on the Like counter. The button provides a one-click shortcut to express a variety of affective responses such as excitement, agreement, compassion, understanding, but also ironic and parodist liking. By asking users to express various affective reactions to web content in the form of a click on a Like button, these reactions can be transformed into a number on the Like counter and are made comparable. Users can materialize their affective responses and Facebook can use them to expand the social graph or count and evaluate them. While the Like button collapses a variety of affective responses, the Like counter combines even further activities such as commenting, sending and sharing into the same metric, since the like is designed as a composite entity as described above.

On top of that, the data produced in such processes are not just metrifications but they also enter various processes of multiplication inside and outside of the platform. First, Facebook advertises the external Like button as a generator of traffic and engagement (Media on Facebook 2010). Likers, the platform argues, are more connected and active than average Facebook users. Each click on a Like button is supposed to lead to more traffic for, and more engagement with, web content, as friends of likers are likely to follow their contacts' recommendations or may be influenced by what their friends like. Engaging with social media, to draw on Grusin (2010), presumes or premediates ongoing interactivity and such an anticipatory climate is facilitated through notification systems highlighting any responses a user has received: "Social networks exist for the purpose of premediating connectivity, by promoting an anticipation that a connection will be made – that somebody will comment on your blog or your Facebook profile or respond to your Tweet" (Grusin 2010, 128). By prompting users to engage with Facebook features on the web and showing what their contacts have engaged with, Social Plugins seek to set a chain of interaction in motion, moving across numerous spaces within and outside the platform. In this context, a like is designed as an ongoing and potentially scalable process. Its value lies both in the present and in the future, in the plus one it adds to the Like counter and the number of x potential more likes, comments, shares or other responses it may generate within the platform. It is in this sense that the infrastructure of the Like economy can be understood as lively (Marres and Weltevrede 2013), as changing internally through the numerous ways in which data are multiplied and content is circulating.

Second, this process of multiplication is based on the creation of differently scaled social formations to which acts of liking, sharing and commenting are being exposed. A series of Social Plugins, for example, are only designed to systematically display activities of particular groups in relation to web content. While the Like counter shows the anonymous number of all likers and sharers, detached from personal profiles, the majority of Social Plugins only depict the activities of a user's contacts and thus looks different to each visitor. Depending on their Facebook privacy settings, a user's click on the Like button may be visible to everyone, to all friends or a selected group of friends and is further distributed across the user's Timeline, different News Feeds and tickers, creating threefold impression statistics for webmasters. If a friend responds to a like with another like or a comment, this activity is exposed to yet another set of users. Each device of the Like economy is creating differently scaled formations of users that are not stable but constantly reconfigured. The data flows between profiles, the exposure on Timelines or News Feeds and the privacy settings allow these formations to scale up to almost every web user or scale down to a selected few Facebook friends. Engagement with web content via Facebook features is thus not only decentralized across a variety of Timelines, News Feeds and tickers, but is also spread across a multiplicity of user formations of different scales.

Through Social Plugins, the previous activities of a user's contacts are presented as potential future activities to users and "delineate a horizon of possibility" to speak with

Langlois et al. (2009), that is create climates in which users are likely to perform some activities rather than others. Although Facebook has recently made an effort to claim that its News Feed, organized through the News Feed algorithm,[148] is not creating an encapsulating echo chamber (Bakshy 2012) or a filter bubble (Bakshy, Messing, and Adamic 2015), the Like economy still features a number of devices that seek to deploy the logic of recommendation cultures in order to set in motion the multiplication of data production. This may be seen in Facebook's mechanism for displaying Related Posts—shown after a user clicks or likes a link to an article in the News Feed—which can then be further liked or shared (Kacholia and Ji 2013).

Third, it is not only user formations and engagement that are being scaled up, the Like economy also contributes to an increasing cross-syndication of content. As mentioned above, with each like or share, web content is being syndicated to different News Feeds, top stories, tickers and Timelines within the platform, thus rendering cross-syndication more scalable. In the framework of the informational web, webmasters produce content to be found. In Facebook's social web, however, content is created to be shared, distributed or cross-syndicated by users to users. In doing so, Facebook is exemplary of sharing as the "distributive and a communicative logic" behind participation in the social web (John 2012, 169). This logic operates within what Jodi Dean refers to as "communicative capitalism" in which the circulation of content has become more important than the message of the content itself (2005, 58). Moreover, users do not have to search for content, but content is presented to them through the multiple recommendation features built into the platform.

As a consequence, a particular relationship between economic value and the social emerges in the case of Facebook, one that is mediated through the creation of connections between the external web and the social graph and through the production of data. User activities are of economic value because they produce valuable user data that can enter multiple relations of exchange and are set up to multiply themselves. However, it is the platform that decides which social activities can be performed and which are turned into comparable and exchangeable data formats, as indicated by the absence of buttons or plugins for critique or disliking.

[148] The News Feed algorithm was previously known as the EdgeRank algorithm. While the original EdgeRank name continues to resonate with the media, researchers and marketeers Facebook itself "hasn't used the word internally for about two-and-a-half years," see http://marketingland.com/edgerank-is-dead-facebooks-news-feed-algorithm-now-has-close-to-100k-weight-factors-55908 [Accessed 24 March 2014]. In an interview with Facebook software engineer Phil Zigoris about Facebook's Promoted Posts, Zigoris states that "EdgeRank is a term that has been used in the past to describe how we optimize the content of news feeds based on what is most interesting to you [as a user]. […] We don't have a product or system called EdgeRank," see http://www.inc.com/howard-greenstein/facebook-promoted-posts-for-small-business.html#ixzz3AjrGBdJM [Accessed 24 March 2014]. Current platform documentation, Facebook's Help Pages, simply calls the much-discussed algorithm the "News Feed algorithm," see: https://www.facebook.com/help/166738576721085 [Accessed 28 July 2014].

Zuckerberg's claim to offer a more social web experience rests on the instant transformation of selected social interaction into forms that can enter further relations in the graph or multiply themselves. Hence, Facebook's very definition of the social web falls together with an increasingly structured, preformatted and traceable web (see chapter 2). Being social online means being traced and contributing to value creation for multiple actors including Facebook and external webmasters. To achieve this, the metrifying capacities of the Like button are inextricable from its multiplying capacities. The medium-specific infrastructure of the Like economy simultaneously enables, measures and multiplies user actions.

Resisting the Like economy

The Like economy thrives on visible and invisible data flows from and to the platform by collecting and exchanging valuable data from users in the form of semi-willing contributions by clicking Like buttons and unwilling contributions through cookies simply by visiting a website with a Facebook feature. Simply logging out, deleting one's profile or not being a member is not enough as independent security consultant blogger Nik Cubrilovic (2011), privacy researcher Arnold Roosendaal (2010), law student Max Schrems (2011) and more recently Belgian researchers Güneş Acar et al. (2015) have shown. Their discoveries and coverage of Facebook's practices have helped to increase the awareness of the issue and have inspired different types of interferences by various actors.

As discussed before, webmasters placing Facebook's social plugins on their websites play an important role in enabling the infrastructure of the Like economy. Privacy-aware webmasters of the German news website *Heise* have developed a new type of Like button that asks users' permission to opt-in before enabling data flows to the platform (Schmidt 2011). *Heise* have developed this button because the original Like button does not comply with the website's data protection and privacy policy. Their two-click Like button, unlike the regular Like button, does not send data to the platform automatically until it has been clicked and activated. This may also be a solution for the German webmasters in the state of Schleswig-Holstein where the Independent Centre for Privacy Protection declared the Facebook social plugins, including the Like button, illegal because they violate the German Tele-media Act (TMG). In August 2011 they ordered webmasters to remove all social plugins from their websites at the risk of a maximum fine of 50,000 euros (Unabhängiges Landeszentrum für Datenschutz Schleswig-Holstein 2011b).[149] On May 22, 2015 *The Wall Street Journal* reported

[149] After commotion in the (German) press about the fine, the Independent Centre for Privacy Protection Schleswig-Holstein released a clarifying statement that this did not immediately concern small website owners and that fines would be proportionally (Unabhängiges Landeszentrum für Datenschutz Schleswig-Holstein 2011a). The Centre is in dialogue with various parties involved, including Facebook, the Federal Trade Commission and the Irish Data Protection Commissioner, and continues to question Facebook's privacy and data practices in light

that the German Consumer Advice Center of North Rhine-Westphalia had sued two major German commercial websites—that had implemented Facebook's Like button—for sending user data to Facebook (Geiger 2015).

Even without the risk of a fine webmasters and bloggers are deciding to remove Like buttons and other social buttons from their sites after their tracking practices became known. On 24 December 2011, software developer and long time blogger Dave Winer declared his blog a "Facebook-free zone" (2011). A blog without a Like button, Facebook comments, or any other plugins connected to Facebook.[150] The choice to remove the button followed Winer's decision to delete his Facebook account after hacker and blogger Nik Cubrilovic had revealed that Facebook was tracking its users on pages with Facebook integrations even when they had logged out (2011).

Besides webmasters, users themselves may also disrupt the data flows by installing special plugins that stop instant data transmissions. These plugins include Facebook Disconnect[151] which is also embedded in the Disconnect[152] plugin that additionally blocks 2000+ other third parties that track you on the web, including other social media platforms such as Twitter and Digg, and search engines such as Google and Yahoo![153] A similar tool is Ghostery, [154] used in the case study in this chapter, which makes most tracking mechanisms visible for users with the option to block them. However, these plugins practices do not protect against new advanced tracking mechanisms such as canvas fingerprinting, a technique which is "hard to detect and resilient to blocking or removing" and employed by one of the most popular social plugins for sharing, AddThis[155] (Acar et al. 2014).

As users are becoming increasingly aware of data-mining practices and privacy issues they are taking the matter of Facebook tracking its users through the Like button higher up. Austrian law student Max Schrems of the "Europe versus Facebook"[156] group has filed 22 official complaints so far with the Office of the Data Protection Commissioner in Ireland where the European headquarters of Facebook are located.[157] He requested his own data with

of the German Tele-media Act (TMG). In 2012 they issued orders against Facebook for its real name policy which violates the TMG (Unabhängiges Landeszentrum für Datenschutz Schleswig-Holstein 2012) and in June 2014 they addressed Facebook's tracking practices with the Schleswig-Holstein Higher Administrative Court (Unabhängiges Landeszentrum für Datenschutz Schleswig-Holstein 2014).

[150] However, as some people note in the comments underneath the blog post, Winer does make use of the third-party commenting system Disqus which allows commenting via Facebook, sharing via Facebook and cross-posting comments to Facebook. This illustrates how difficult it can be for webmasters to turn their blogs into a completely Facebook-free zone.

[151] See: https://chrome.google.com/webstore/detail/facebook-disconnect/ejpepffjfmamnambagiibghpglaidiec/details?hl=en [Accessed 22 July 2014].

[152] See: https://disconnect.me/disconnect [Accessed 22 July 2014].

[153] See: https://disconnect.me/ [Accessed 23 July 2014].

[154] See: https://www.ghostery.com/ [Accessed 22 July 2014].

[155] AddThis offers a set of social buttons for sharing so that webmasters only have to implement 1 set for displaying multiple social buttons at once.

[156] See: http://www.europe-v-facebook.org/ [Accessed 22 July 2014].

[157] For an overview of the complaints, see http://www.europe-v-facebook.org/EN/Complaints/complaints.html [Accessed 22 July 2014]. Europe versus Facebook is currently preparing the largest privacy class action in Europe

Facebook Ireland under European law and he received a file containing 1,222 pages. When going through the documents he noticed that the file included deleted information but did not include his own Like data.[158] Upon a second request Facebook responded that it would not provide "any information to you which is a trade secret or intellectual property of Facebook Ireland Limited or its licensors."[159] Likes are considered valuable proprietary data within the Like economy and do not belong to the user, even when requested under European law.

Various consumer and privacy groups in the US have requested the Federal Trade Commission to investigate Facebook's tracking, and in November 2011 Facebook settled a privacy complaint with the FTC. The settlement describes the privacy promises Facebook did not keep and the consequences of the proposed settlement which

> bars Facebook from making any further deceptive privacy claims, requires that the company get consumers' approval before it changes the way it shares their data, and requires that it obtain periodic assessments of its privacy practices by independent, third-party auditors for the next 20 years (FTC 2011).

In addition, concerned Californian citizens have filed multiple class action complaints against Facebook for violating privacy laws, which include tracking users using the Like button and using people's names and photos for recommendations when clicking a Like button.[160]

Another type of interference to disable or disrupt the data flows between websites, users and Facebook takes place in the form of artistic interventions. The FB Resistance group featured a script by @xuv that automatically likes all your friends' updates "if you're too busy to show them love manually" (Deswaef 2014) which not only subverts the idea of liking but at the same time adds noise to Facebook's valuable data flows. *Facebook Demetricator* by artist and researcher Benjamin Grosser is another project which directly engages with Facebook's web economy of metrification. *Facebook Demetricator* is a "web browser extension that hides all the metrics on Facebook" (Grosser n.d.). It interferes with the mechanisms of the Like economy—which transform various forms of engagement into quantified Likes—by removing all the numbers from the interface (see figure 22).

against Facebook Ireland, see: http://europe-v-facebook.org/EN/Complaints/Class_Action/class_action.html [Accessed 28 April 2015].

[158] See: http://www.europe-v-facebook.org/EN/Data_Pool/data_pool.html [Accessed 22 July 2014].

[159] See: http://www.europe-v-Facebook.org/FB_E-Mails_28_9_11.pdf [Accessed 22 July 2014].

[160] See: http://law.justia.com/cases/federal/district-courts/arkansas/arwdce/5:2011cv05266/38516/1/ and http://dockets.justia.com/docket/california/candce/5:2013cv05996/273216 [Accessed 23 July 2014]. A settlement in one of these cases required Facebook to change its policy for the Like button, see: http://www.nytimes.com/2012/06/22/technology/to-settle-suit-facebook-alters-policies-for-like-button.html [Accessed 23 July 2014].

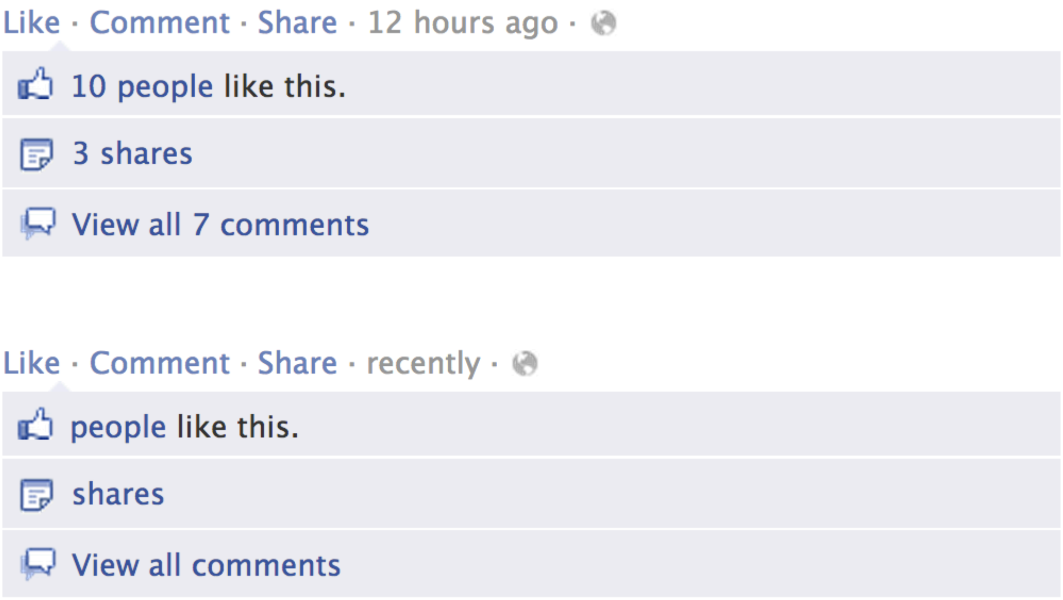


Figure 22: The Facebook Demetricator browser plugin. The plugin hides all the metrics on Facebook. It removes the number of likes, shares, tweets and comments as well as the timestamps in the News Feed. Original Facebook News Feed (top), demetricated Facebook News Feed (bottom). © Benjamin Grosser. 2012. Image reproduced with permission from the artist.

The browser plugin eliminates the metrics in the user interface in an attempt to eradicate their quantifying effects on users. *Facebook Demetricator* removes the visibility of likes in the front-end, the interface, and not the actual likes in Facebook's back-end, the database. As such it can be seen as a form of critical engineering, which sets out to unveil the politics of engineering—the design and build of a system—by studying and exploiting its language of engineering (Oliver, Savičić, and Vasiliev 2011; Gehl 2014). Grosser's *Facebook Demetricator* exposes how Facebook has been engineered as a platform based on the quantification of affect. An important aspect of critical engineering is "to determine methods of influence and their specific effects" (Oliver, Savičić, and Vasiliev 2011) where Benjamin Grosser shows the consequences of quantification on user behavior by removing the metrics.

Grosser actively collects user feedback on the tool to understand the effects metrification. This feedback shows how users experience the multiplication of data production in the Like economy and "illuminates how metrics activate the 'desire for *more*,' driving users to want *more* 'likes,' *more* comments, and *more* friends" (Grosser 2014). This desire for more, as argued by Taina Bucher, is also engrained in the algorithmic logic of Facebook which is engineered to make items with more interaction more visible (2012c, 1174). Bucher describes how Facebook rewards participation with algorithmically-determined visibility in the newsfeed and this "making it appear as if everybody is participating and communicating by emphasizing those stories that generate many Comments and Likes provides an incentive to Like or Comment as well" (2012c, 1175). In his design of the *Demetricator* tool artist Benjamin Grosser interferes in this process by removing the visible numeric incentive to participate with the aim to "disrupt the prescribed sociality produced through metrics, enabling a social media culture less dependent on quantification" (2014).

In the above part I have addressed how various actors have critically engaged with the data-intensive infrastructure of the Like economy through a series of interferences. These vary from taking Facebook to court, to blocking data flows from external websites to Facebook, to adding noise to the Like economy, to disabling the infrastructure's multiplication processes to disrupt Facebook's quantified sociality. To conclude, in the final part of this chapter I address the limits of sociality and data production in Facebook's Like economy.

The limits of sociality in Facebook's social web

The Like economy can be understood as part of emerging free economies which offer services for free and generate profits via their by-products (Anderson 2009)—in Facebook's case social activities. Corporate interest in social interactivity and user affects are not new to Facebook, but have to be understood in the trajectory of information-intensive post-Fordist economies, corporate interest in transactional online data, and attempts to objectify consumer affects (Arvidsson 2011). The increasing centrality of knowledge and information in post-Fordist modes of production (Thrift 2005) have contributed to a further intermingling between life and production, between social interactivity and economic value and it is especially the web that provides particular infrastructures to cater for these interdependences. In particular social media platforms are engineered as such to convert web activities in the front-end, into valuable data in the back-end (Stalder 2012; Gehl 2014).

In the informational web, user preferences and basic activities could be read from server log files, which are used to derive engagement measures such as hits and time spent on a page. With the rise of the social web, companies realized that everyday online activities provide a rich source of information about user preferences, habits and affects that had previously only been available through consumer research techniques. An increasing range of social media monitoring services is currently tracking and analyzing user behavior online, instantly turning social activity and web engagement into different types of data (Lury and Moor 2010). Facebook's endeavors are thus not new and have to be accounted for in the context of corporate social media monitoring. However, the Like economy creates an infrastructure that not only allows transactional data to be mined instantly, but also allows it to be attached to individual user profiles and multiplied.

Throughout this chapter I have approached this development from a medium-specific perspective by discussing how web devices such as social buttons and the Open Graph have contributed to this intermingling of the social and data and their multiplication. In comparing the emerging Like economy with the hit and link economy, I explored how the launch of social buttons has reintroduced the role of users in organizing web content and the fabric of the web—and how the infrastructure of the Open Graph is turning user affects and engagement into both data and objects of exchange in the platform-specific currency of the Like. This chapter has presented a twofold analysis of the Like economy.

First, by showing how Facebook is creating a particular fabric of the web through social buttons, which at the same time decentralizes data collection and recentralizes data processing and economic valorization. I traced the presence of Facebook Connect and Social Plugins across the web, showing that Facebook is currently emerging as a key agent in the sector of back-end data mining. More than that, such data mining practices reveal an alternative fabric of the web, one that is not organized through hyperlinks placed by webmasters, but one that is based on data flows enabled by and to third-party devices, facilitating the decentralization of data mining and the recentralization of data processing within platforms.

Second, by following the medium-specific perspective further, I have drawn attention to the capacity of the Like button to metrify user affect and engagement by turning them into numbers on the Like counter while strategically exposing them to other users to evoke further interactions. User activity on social media platforms has so far often been discussed in a post-Marxist terminology of labor, production and user exploitation. It has been understood as a form of social production (Scholz and Hartzog 2007), as presumption (Ritzer and Jurgenson 2010) or working consumers/users (Fuchs 2010) or as free labor (Terranova 2000), as users voluntarily engage in productive activities without financial reward for their contributions. The Like economy cuts across too simplistic ideas of exploitation by establishing webmasters, who place these social buttons on their sites, as intermediary actors. The medium-specific perspective draws attention to the role of devices, as affect is not valuable per se, because it is hard to measure and to compare. It is the medium-specific infrastructure of the Like economy that allows the transformation into quantified likes, which can then enter multiple forms of exchange: from producing data for user mining and patterning, to creating recommendation traffic from Facebook, providing access to Like button statistics or moving behind the Likewall. This medium-specific infrastructure further creates an environment that does not require active participation in the Like economy through clicking on social buttons or commenting. Instead, the underlying data mining processes foster participation by default, tracking users' browsing behavior, storing Like button impressions or instantly sharing app engagement to the ticker. By looking at the infrastructure that enables these processes attention is drawn to the politics of the Facebook platform and its back-end data flows in which logging out, deleting one's profile or never joining the platform do not offer solutions to opt out.

To conclude, I return to Zuckerberg's ambition to integrate ever more social activities into the Facebook platform. As former employee Matt Cohler claims: "Facebook has always thought that anything that is social in the world should be social online" (Gelles 2010). In contrast, I would like to outline that there are limits to Facebook's enclosure of sociality, most notably in the current absence of the widely requested Dislike button as a critical counterpart to the Like button. Although such a button may comply with corporate interest in both positive and negative sentiment, Facebook abstains from its implementation, as a one-click solution for negative affect may lead to insensitive use (Sawers 2010). Yet, the decision to refrain from "disliking" also bears economic dimensions as opening up the possibility for controversial sharing practices would create negative traffic and negative connections, which cannot be collapsed in the composite Like counter. In this sense, the Like economy is facilitating a web of positive sentiment in which users are constantly prompted to like, enjoy, recommend and buy as opposed to discuss or critique—making all forms of engagement more comparable but also more sellable to webmasters, brands and advertisers. While Social Plugins allow materializing and measure positive affect, critique and discontent with external web content remain largely intensive and non-measurable. The absence of negative affects has until the Spring of 2015 marked the limits of Facebook's understanding of sociality. The introduction of new activity apps, however, has complicated the affective space of Facebook, potentially allowing for differentiated and even negative activities in relation to web objects, such as hating, disagreeing and criticizing[161]—while the action "dislike" remains blocked.

The Like economy has thus created an infrastructure that comes across as facilitating a more social web experience, but participates in creating an alternative fabric of the web in the back-end, one in which social interaction is instantly metrified and multiplied and which connects insights from web analytics with individual user profiles and the social graph. It enables only particular forms of social engagement and creates specific relations between the social, the traceable and the marketable, filtering them for positive and scalable affects.

In this chapter I have analyzed how Facebook has created a data-intensive infrastructure on the web by using its platform-specific features, in particular the Like button. This infrastructure is characterized by the double logic of platformization, that is the decentralization of data production and the recentralization of these processes by sending valuable data back to the platform. To expand further into the web Facebook employs webmasters to open up their websites to Social Plugins to enable data flows from and to the platform. These Social Plugins not only permeate the boundaries of the platform (see chapter 2) but also the boundaries of the website by establishing channels for data exchanges (see chapter 6). In this sense platforms and websites become interconnected and mutually shape each other through web activities taking place across various spaces.

The next chapter addresses this permeability of the website in the web as platform as a consequence of the integration of third-party content and functionality. Rather than seeing the website as a concrete entity or a bounded object, I argue for reconceptualizing the study of websites as website ecology. The website as an ecosystem inhabited by third parties forms the entry point to map and analyze the larger techno-commercial configurations of the web that

[161] However, all apps that want to make use of the pre-defined actions by Facebook or create their own custom actions must be reviewed and approved by Facebook first, see: https://developers.facebook.com/docs/sharing/opengraph/using-actions [Accessed 1 May 2015].

these websites are embedded in. In this chapter I have provided a method to map the larger tracking ecosystem of Facebook, or the platformization of the web by Facebook. In the following chapter I further expand this method to map tracking ecologies over time. I position the source code of the archived website as an important entry point to reconstruct these historical website ecologies.

6. Website ecologies: Redrawing the boundaries of a website

In the previous chapters I have shown how websites (chapter 2 and 5) and blogs (chapter 3) in the platformized web are increasingly assembled from various third-party sources such as social media platforms. Websites are no longer discrete entities but have become shaped by and entangled in relationships with external actors. The notion of the data pour (chapter 2) has drawn our attention to seeing the website as an assembled object. In this chapter I build on the previous chapters with a historical perspective on the changing composition and boundaries of the website. I propose to see the website as an ecosystem through which we can analyze the larger techno-commercial configurations that websites are embedded in. In doing so, I reconceptualize the study of websites as website ecology. The website's ecosystem can be detected by examining the source code in which a website's connections with third parties have become inscribed. Complimentary to this, I explore how we can examine changes in a website's ecosystem *over time* as a way to trace the platformization of the web at large through the changing composition of the website.

I position the historical study of a website's ecosystem as a contribution to web historiography, a sub-field of internet history, which is concerned with writing histories of the web (Brügger 2009; Kirsten Foot and Schneider 2010; Ankerson 2012; Brügger 2013). Analyzing a website's changing composition over time can provide us with new insights into the changing techno-commercial environments of the web and the spread of platformization. To operationalize this contribution I turn to web archives, which serve as important tools for web historians who wish to uncover previous states of the web. I argue that while the web archive of the Internet Archive Wayback Machine focuses on snapshots of single websites (Brügger 2009; Rogers 2013; Ben-David and Huurdeman 2014) the source code of these snapshots contains valuable information about a website's relations with third parties that we can employ for the reconstruction of historical website ecosystems and missing platform content. That is, I propose treating the source code as a demarcation object that determines the dynamic interrelations between websites and external actors, by focusing on the code snippets of third-party objects that enable these connections. Additionally, while embedded third-party content or functionality from social media platforms may not be included in the archived website due to the archiving process, the embed code or data pour that refers to them is often still present in the archived source code and can be used to retrieve missing platform content. In two case studies I will examine how we can employ archived source code to reconstruct historical website ecologies and to retrieve missing social media platform content in archived websites. In the first case study I develop a method to trace one particular type of third-party object in the archived source code, the tracker, to analyze the historical tracking ecologies that the New York Times website has been embedded in between 1996-2011. In the second case study I address the problem of missing third-party content in archived websites by developing a method to reconstruct the missing commenting space of an archived Huffington Post article. In this study I make use of the code snippets of Facebook Comments' plugin in the archived source code to retrieve all the comments from Facebook's database that have not been archived.

In this chapter I argue that the platformization of the web has contributed to the demise of the website as a bounded object and propose to reconceptualize the study of websites as website ecology.

Website ecology

In the social web third-party content and functionality are substantially shaping websites (Mayer and Mitchell 2012). Webmasters can make use of social media platforms to embed sharing functionality, related social activities such as tweets, commenting systems to enable and manage comments, content delivery networks to host videos and photos as well as advertising servers to display dynamically-generated personalized ads to generate income. In these scenarios, the website is no longer a self-contained unit but has become informed and molded by other actors on the web.

Thus, I would like to introduce the notion of website ecology, that is the study of the complex socio-technical relations between websites, users, social media platforms, tracking companies, other actors and their environment. Chi et al. use the term web ecology to argue that the complex set of relations between web content, their users and the structure they create between content "form an *ecology* among users and their information environment, and its change through time is a form of evolution" (1998, 400). In this dissertation I use the term in a similar manner but at the same time I recognize that the types of web content and the relations that can be created between content and users have changed significantly compared with the web Chi et al. previously described. Most significantly, relationships between web content have moved beyond mere hyperlinks between web pages so as to include other relationship markers that have been created through the use of social media platform features (see chapter 3, 4 and 5). In this chapter I employ the term website ecology to draw attention to the techno-commercial environments that websites are embedded in through a set of complex relations with third parties.

In doing so, I draw parallels with media ecology. Neil Postman defines media ecology as "the study of media as environments" which

> looks into the matter of how media of communication affect human perception, understanding, feeling, and value; and how our interaction with media facilitates or impedes our chances of survival. The word ecology implies the study of environments: their structure, content, and impact on people (Postman 1970).

Shifting media ecology's focus away from studying the effects of media on people towards the materiality of these media environments, Matthew Fuller argues for understanding media ecology as "the massive and dynamic interrelation of processes and objects, beings and things, patterns and matter" (2005). Here, I am particularly interested in reflecting on media ecology after the "softwarization" of media (Berry 2012; Manovich 2013, 5). In *Software Takes Command* new media scholar Lev Manovich reconceptualizes media ecology after our "media becomes software" (2013, 156). For Manovich media ecology refers to the study of the software environment of media data and its distinct software techniques (2013, 150). Software studies scholar David Berry, on the other hand, does not use the term "media ecology" to describe our software-saturated media environment but describes our current media system as a "computational ecology" which is comprised of distinct "software ecologies" (2012). Berry employs the term ecology as "as a broad concept related to the environmental habitus of both human and non-human actors" (2012). Similarly, José van Dijck too draws from ecological metaphors to study software-mediated spaces on the web. She analyzes the interconnectedness of social media platforms as an "ecosystem of connective media" where each platform is seen as a microsystem (2013c, 21).

Following these authors, I draw from ecology in a similar manner to analyze changes in the composition of the web by studying the relations between a website and its environment.[162] Website ecology looks at the dynamic and shifting relations between websites and third parties, which do not only become interconnected through users' web activities such as linking pages, but also through software, platform features and data flows.

As discussed in the previous chapters, the programmability and modularity of websites have enabled platform ecologies in which data flows in structured exchanges between websites, users and platforms, as well as third parties such as advertisers, tracking companies and social media aggregators. An ecological approach to understanding websites allows for the analysis of websites as dynamic spaces where these complex relations between users, websites and third parties get encoded.

Previous approaches to studying the website in its networked environment have focused on how websites establish relations with other websites through linking, therewith embedding the website in a hyperlink network (Park 2003; Rogers 2002). Elmer and Langlois describe these approaches as "Web 1.0 methods focused on mapping hyperlink networks" (2013, 43) to analyze the connections between websites and the networks they form. Previously Elmer has called for detecting new indicators of networking and to develop "a broader vision for the analysis of web code, expanding beyond the mapping of HREF tags (hyperlink code) toward an understanding of the larger structure and deployment of all web code and content (including text, images, met tags, robot.txt commands and so on)" (2006, 9). In this chapter I

[162] I am aware of the complications of drawing from biological metaphors since some "biological laws and principles do not allow a technological translation" (Scolari 2012, 218). However, Carlos Scolari and Marianne van den Boomen argue, metaphors play an important role in conceptualizing new media technologies (2012, 206; 2014) and in expanding media studies theory (Scolari 2012, 2010).[162] In this dissertation, the ecological metaphors function as conceptual devices to make sense of the new complex arrangements between humans and software spaces on the web by way of an analogy.

contribute a new approach that examines the source code for dynamic third-party scripts that create connections with third parties.

Web 2.0, Elmer and Langlois argue, is characterized by new forms of networked connectivity which move beyond the hyperlink and which require new methods to map these new types of connections (2013, 44). They outline the "building blocks" of what they refer to as Web 2.0 "cross platform based methods" in which they trace objects across platforms to detect their channels of circulation and analyze the different types of relationships that they form (2013, 45). Here, I build on Elmer and Langlois' idea of cross platform analysis with a novel method that traces cross connections from within the website by employing third-party objects.

A cross platform approach shifts the attention away from the hyperlink as the prime connection mechanism towards other web-native objects, or platform-specific objects such as social buttons, that enable and mediate interactions between a website and its ecosystem (see chapter 4 and 5). Of particular interest here are the objects that are not immediately visible in the front-end, that is the user interface, and which create relations with other actors on the web such as trackers. Roesner et al. define a (third-party) tracker as "a website (like doubleclick.net) that has its tracking code included or embedded in another site (like cnn.com)" to "identify and collect information about users" (2012, 12). Trackers can be divided in first party trackers and third-party trackers. First party trackers are issued from the same domain as the website "that the user has voluntarily interacted with" whilst third-party trackers are issued from a different domain than the website, indicating involuntary interactions with an external actor (Mayer and Mitchell 2012, 413). In this chapter a tracker refers to a third-party tracker, indicating a connection with an external actor.

Web bugs, beacons and other types of trackers embed the website in larger techno-commercial configurations on the web by establishing relations between websites and advertising networks, analytics companies and market research companies amongst others. Detecting these relations requires moving beyond the user interface (Langlois, McKelvey, et al. 2009) and looking beneath the surface of a website in order to detect the traces of these dynamic relations by engaging with the materiality of a website, the source code.

In this chapter, I draw from European media ecology[163] which emphasizes the materiality of code and software of our contemporary media environment (Fuller 2005; Goddard and Parikka 2011; Berry 2012). In this view, the source code of a website forms the object of the study of website ecology. The website's source code provides the material in which the relations with other actors become inscribed through dynamic third-party content, objects and

[163] For the differences between the North American version of media ecology, as introduced by Neil Postman and Marshall McLuhan and others, and the European version by Matthew Fuller, Jussi Parikka and others, see Michael Goddard (2011). This essay is part of a special *Fibreculture* issue on 'Unnatural Ecologies' which, as editors Michael Goddard and Jussi Parikka outline in their introduction, reflects on media ecology as a direction within media studies with its revival with Matthew Fuller's new take on it in *Media Ecologies: Materialist Energies in Art and Technoculture* (2005).

features. This follows a perspective advocated by a number of authors (Manovich 2001; Hayles 2004; Fuller 2005; De Souza, Froehlich, and Dourish 2005; Mackenzie 2006; Marino 2006; Kirschenbaum 2007; Brügger 2008; Ankerson 2012; Berry 2012) who engage with the materiality of new media by pointing to the source code as an important entry point for analysis.

Within these approaches, I am particularly inspired by De Souza et al. who propose to see the source code as "a social and technical artifact" in which aspects of software development have become inscribed (2005, 197). They draw from Latour's notion of inscription (1999)[164] to refer to "a process through which social practice and technological artifacts become inextricably intertwined" (2005, 197). They see "software artifacts as pure inscriptions" that can be used "to uncover the structure of software projects" and their development processes (2005, 197), an approach they refer to as "an 'archeology' of software processes" (2005, 206). Similarly, Megan Sapnar Ankerson turns to the traces of software to engage with "the culture of software in constructing histories of the web" thereby bringing a "software studies lens to web historiography" (2009, 195). I build on De Souza et al.'s and Ankerson's approaches by seeing the archived source code as a document in which relations with third parties become inscribed that can be used to reconstruct historical techno-commercial configurations on the web.

The source code reveals dynamic third-party scripts and objects—the data pours—that enable connections with external databases. In this sense this chapter can also be seen as a contribution to Critical Code Studies (CCS), an approach aligned with software studies and platform studies which analyzes the code layer of software (Marino 2014). CCS is mostly concerned with analyzing artistic programs and code poetry, yet is open to the interpretation to any kind of code (Marino 2006). I specifically address a concern expressed by Matthew Kirschenbaum during the 2011 HASTAC Scholars Critical Code Studies Forum that "by focusing on the analysis of code snippets, CCS could potentially abstract code from its larger software constructs" (2014). The approach put forward here uses code to direct the attention back to these larger software constructs of the web such as tracking ecologies.

Tracking ecosystems

Trackers in the form of beacons and analytics can be intentionally implemented by webmasters to monitor the functioning of their websites or collect data about their visitors. However, as demonstrated previously in chapter 5 on Facebook's Like Button, trackers can also come as a by-product of third-party functionality. Webmasters that employ website

[164] Latour defines inscription as "a general term that refers to all the types of transformations through which an entity becomes materialized into a sign, an archive, a document, a piece of paper, a trace" (1999, 306).

analytics, advertisements or social buttons therewith—intentionally or unintentionally—embed the website and its visitors in a tracking ecosystem.[165]

Since the early days of the web, banner ads, cookies and other tracking web objects have been an integral part of the web. In October 1994, only a year after the release of the first graphical browser Mosaic, HotWired placed the first banner ads on its website which marked an important turning point for the web (D'Angelo 2009). Another second development was the creation of ad networks connecting advertisers to webmasters (Gehl 2014, 104). According to Tim O'Reilly, ad network DoubleClick—founded in 1996—was a "pioneer[…] in treating the web as a platform:"

> People don't often think of it as "web services", but in fact, ad serving was the first widely deployed web service, and the first widely deployed "mashup" (to use another term that has gained currency of late). Every banner ad is served as a seamless cooperation between two websites, delivering an integrated page to a reader on yet another computer (O'Reilly 2005).

DoubleClick 's ad serving technology is offered as software as a service (SaaS),[166] enabling real-time communications between websites hosting ads and the ad network in order to retrieve the most relevant banner ad matching the website's visitor (Peters 1999, 342). The banner ad creates a data pour for two-way data flows between websites and DoubleClick's ad network.

A third development came from ad networks such as Google AdSense which started selling ads on small websites such as personal homepages and blogs in contrast to DoubleClick which mainly sold ads on major websites such as web portals (O'Reilly 2005; Gehl 2014, 104). Chris Anderson argues that Google makes most of its money from ads on small websites, which he refers to as "the long tail of advertising" (2006, 24). These ad networks distribute their ads across websites and recentralize user data collected from these websites back to these services, which can be seen as an early example of the dual logic of platformization. With the rise of ad servers to track and monitor ads on third-party websites, trackers have become an integral part of the interactions between websites and their ecosystem (Mayer and Mitchell 2012). Next, I will discuss this changing composition of the website within the web as platform.

The website as an assembled unit

In the early days of the web, often referred to as Web 1.0, websites were considered to be fairly self-contained units since most content was stored on the same server (Song 2010, 251;

[165] In chapter 5 I have mapped the tracking ecosystem of the Alexa top 1000 websites, see figure 25.

[166] SaaS, or "Software as a Service" can be defined as "[s]oftware applications that are hosted on the internet and delivered on demand, through a web browser. SaaS is the Web 2.0 answer to the traditional software model of shrink- wrapped products, installed on a user's PC or on the local network" (Funk 2008, 5).

Mayer and Mitchell 2012). Within Web 2.0 websites are increasingly entangled in a networked context and shaped by third-party content and dynamically-generated functionality (Mayer and Mitchell 2012; Gehl 2014, 103).[167]

The modularity of websites and their integration of data pours are keys to understanding the altering role and shape of the website in the web as platform. The changing nature of the website, Alan Liu argues, can be understood as shifting from web pages as self-contained units in Web 1.0 to webpages which "increasingly surrender their soul to data pours that throw transcendental information onto the page from database or XML sources reposed far in the background" as exemplary within Web 2.0 (see chapter 2). Data pours have interrupted "the page-based paradigm of the web" by "abstract[ing] web content into feeds, real-time flows of XML data" (Whitelaw 2008) and, I argue, have contributed to the website's demise as a bounded object. Robert Gehl similarly argues that in Web 2.0 a website is assembled from third-party sources: "a website is a 'mash-up' of top-down, incrementally altered architecture, bottom-up user participation and processing, and the lateral insertion of advertising, creating a coherent visual artifact out of these different streams" (Gehl 2014, 103).

Within the web as platform the website can be seen as an assemblage of modular elements that on the one hand enable interactions with other actors on the web and on the other hand permeate or redraw the boundaries of the website by setting up data pours—channels for the exchange of content and data stored in external databases. Next, I will show that the website as an assembled object provides an important entry point for analyzing historical website ecosystems through web archives.

The detachment of the website ecosystem

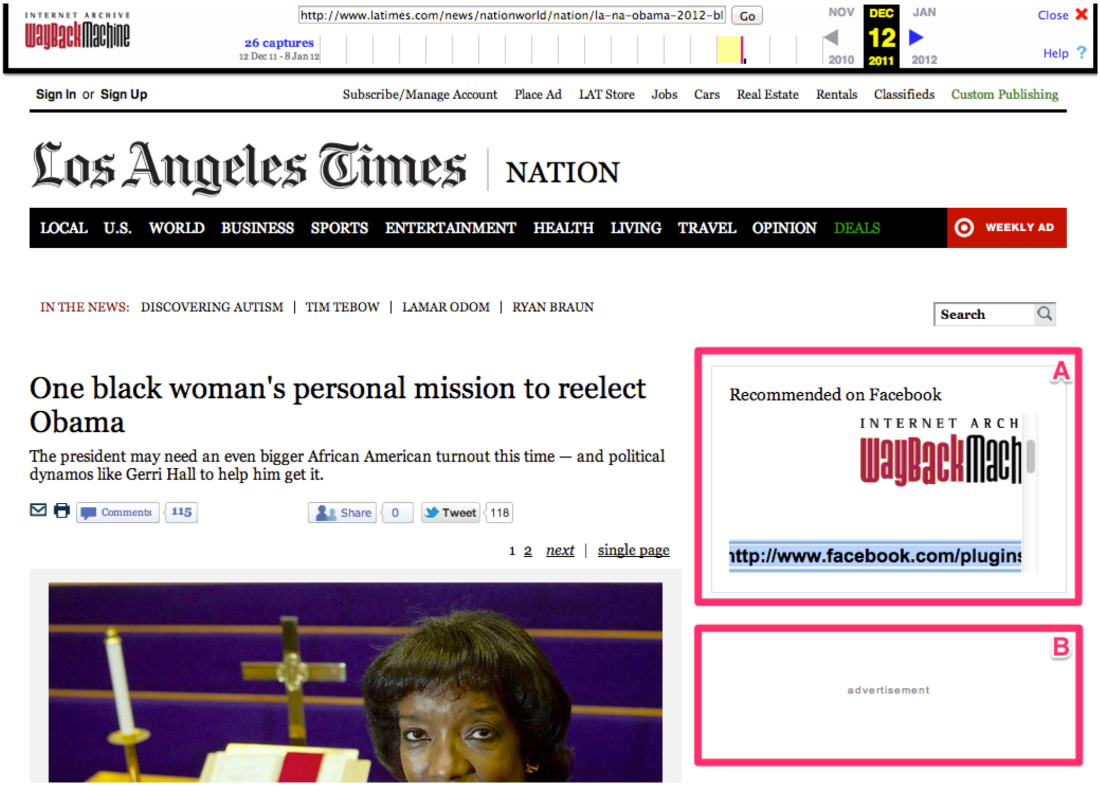
To study previous states of the web, the web historian needs access to historical material which can be found in web archives such as the Internet Archive Wayback Machine. However, web historian Niels Brügger argues, the archiving process actively shapes and determines how a website is archived and therefore what kind of reconstruction or historical analysis is possible (Brügger 2009, 126). Brügger distinguishes between five interrelated analytical web objects on the web:

> Whether studying the web of today or of the past, we can focus on five different web strata: a web element, for example an image on a webpage; a webpage is what we see in a browser window; the website is a number of coherent webpages; the web sphere is the web activity related to a theme, an event or the like; and the web as a whole is anything that transcends the web, such as the general technical infrastructure of the web or the content of the web in its totality (Brügger 2013, 753–754; Brügger 2009, 122–125)

[167] As I have previously addressed in the introduction, I do not mean to imply a radical break between websites in Web 1.0 and Web 2.0. Rather, the previous example of DoubleClick serving ad banners in early websites shows the continuities of these practices.

Brügger describes that the "impact of the archiving process on the archived web material is that in practice the website is almost always the basic unit in a web archive" (2013, p.757). He defines the website as a mediated technical artifact which is "a coherent textual unit that unfolds in one or more interrelated browser windows, the coherence of which is based on semantic, formal and physically performative interrelations" (2009, 126). Web historians Foot and Schneider delineate websites as "groups of pages sharing a common portion of their URL" (2010, 69) where web pages are seen as "groups of elements assembled by a producer and displayed upon request to a server" (2010, 69). These definitions of the website as an object of study focus on the visible rendering of the website as a coherent yet assembled object. What we see in the archiving process is that the increasing modularity of the website poses a challenge to assembling the archived website but also that the archived website is detached from its larger context.

This may be seen in the archived websites in the Internet Archive Wayback Machine, one of the largest available web archives. As previously discussed in chapter 3, the Internet Archive Wayback Machine, Rogers argues, lends itself to "single-site histories" or "website biographies" as one accesses the archive by entering a single URL, the website's domain name (2013, 66). The focus on the single website shows how in the process of archiving, the website has been separated from its ecosystem. The foregrounding of the website over the other web strata means that archives often privilege the content of a website over "the search engine results that once returned them, the references contained in them (hyperlinks), the ecology in which they may or may not thrive (the sphere), and the pages or accounts contained therein that keep the user actively grooming his or her online profile and status (the platform)" (Rogers 2013, 63). Thus, in the archiving process the website is detached from the techno-social context it resides in (Weltevrede 2009, 84). This detachment occurs in different manners, think for example, of the website's log file which contains information about its visitors, its website statistics, its ranking on Alexa, its advertisements and its integrated Facebook recommendations and comments (see figure 23). Figure 23 illustrates how, in addition to being detached from its context, the archived website is also missing the third-party content and functionality shaping the website, an issue that will be addressed in the second case study.



One black woman's personal mission to reelect Obama

The president may need an even bigger African American turnout this time — and political dynamos like Gerri Hall to help him get it.

Figure 23: Archived webpage from the LA Times in the Internet Archive's Wayback Machine with missing advertisement and missing Facebook recommendations. The screenshot highlights the missing Facebook plugin and missing advertisement. Retrieved from: https://web.archive.org/web/20111222065750/http://articles.latimes.com/2011/dec/12/nation/la-na-obama-2012-black-voters-20111212 [Accessed 3 December 2012].

While the website is the main unit within web archives (Brügger 2013, 756), the archived website's source code also contains elements that can be employed to "uncover parts of the web that were not preserved" (Samar et al. 2014, 1199). In what follows next I develop a novel method that moves beyond the single-site history by employing the code snippets of an archived website to reconstruct a website's ecosystem. My method addresses the conceptual and practical problem of the website as a bounded object which has troubled web archiving theorists (cf. Schneider and Foot 2004; cf. Brügger 2009). In shifting the focus from the content of the archived website to the code (cf. Rogers 2013, 63–64), different analytical opportunities present themselves. The archived website with its missing content can be considered incomplete, however when turning to the source code of the archived website it is more complete than it initially seemed.

Analyzing historical website ecosystems

In the source code of archived websites we can find the traces,[168] the code snippets, of web objects such as trackers and third-party content. While these objects themselves may not be archived their code allows for reconstructing the network of trackers that websites have been embedded in.

Previous approaches in using aspects of a website's source code to study its environment include outlinks to other websites as one way to move beyond the single-site history. As shown in chapter 3, the HTML code for hyperlinks in an archived website's source code enables the reconstruction of past hyperlink networks through a historical hyperlink analysis.[169] Even though these websites may not have been archived themselves, the outlinks pointing to them allows for "conjuring up" these websites (Stevenson 2010b) and map past states of the web or the blogosphere using the Internet Archive (Stevenson 2010b; Ammann 2011; Ben-David 2011) (see chapter 3). Following Elmer (2006) and Elmer and Langlois' (2013) call for employing other web-native objects for networking, I move beyond the hyperlink and focus on trackers as objects that entangle the website in a techno-commercial web environment.

Previous historical tracker studies include a longitudal study on trackers on 1200 websites between October 2005 and September 2008 (Krishnamurthy and Wills 2009) and cookies and their (default) settings in different Netscape browser versions over time (Elmer 2002). While web archiving has taken the first steps to attend to websites as part of hyperlink networks, little attention has been paid to the historical study of tracker networks so far.

My proposed methodology to analyze historical tracking networks builds on previous research to map tracking networks (see chapter 5). In this chapter I further extend this method to detect trackers in *archived websites* to analyze the tracking networks that websites have been embedded in over time. At the same time this methodology serves as a blueprint for the development of further methods that focus on detecting and mapping other third-party objects in archived websites to study previous states of the web through website features and technologies and to examine the platformization of the web at large.

The methodology to create tracker networks is inspired by a digital methods approach of "repurposing" the existing analytical capacities of tools and devices for research with the web (Rogers 2013, 1). Many tools on the web have a methodological approach built into them to achieve a particular functionality. An example of such a tool is the browser add-on Ghostery, which

[168] Following Kirschenbaum (2003) and Ankerson (2009) I use the word traces to refer to the material evidence of software, in this case the tracker code.

[169] Despite the fact that web archives are often incomplete, the proposed method does not require the linked website to be archived as well. The mere presence of the hyperlink pointing to it can be used to map the historical hyperlink network of a blogosphere as demonstrated in chapter 3.

> scans the page for trackers - scripts, pixels, and other elements - and notifies you of the companies whose code is present on the webpage you are visiting. These trackers often aren't otherwise visible and are often not detailed in the page source code. Ghostery allows you to learn more about these companies and their practices, and block the page elements from loading if the user chooses." (Ghostery n.d.)

Ghostery has an inbuilt method to detect trackers in websites that can be employed for research purposes. Instead of creating a new method or tool to find trackers, we can also "repurpose" the existing analytical capacities of this existing tool. In the project "Tracking the Trackers" (2012),[170] colleagues and I repurposed Ghostery to analyze and map the presence of trackers in a collection of websites (see chapter 5).

Ghostery looks for patterns of trackers and matches them to a database of over 2000 known trackers.[171] It uses simple string matching (matching a number of characters in a code string) and regex as a method to detect and match the found tracker code against their database of trackers.[172] For example, Ghostery looks for the presence of advertiser DoubleClick on a website by examining the website's source code for known DoubleClick patterns in Ghostery's tracker database, for example [ad.doubleclick.net] or [doubleclick.net/pagead]).

The main contribution of repurposing Ghostery—by building a new tool on top of it—is the achievement to detect and map *tracker networks*. While Ghostery has been developed as a plugin to detect and block trackers on an *individual website*, the Tracker Tracker tool is able to detect trackers in *a collection of websites* and to create *a network view* of websites and their trackers. The Tracker Tracker tool can scan up to a 100 URLs at a time and outputs the name of the website, the tracker found, the tracker pattern and the tracker type in a .csv spreadsheet and .gefx file. This latter file, a Gephi graph, also contains the relations between the trackers and the collection of websites, based on tracker presence, and can be used to visualize the network of websites and their trackers using the graph visualization tool Gephi.[173] Scanning large collections of websites for trackers enables mapping of tracking networks that websites are embedded in (see figure 21 in chapter 5).

Such an approach no longer focuses on the relations between websites, by reconstructing a network based on the visible outlinks found on the website, but instead it focuses on the invisible connections established with trackers such as central ad servers and platforms in the back-end. This approach, which looks at alternative devices to organize relations on the web,

[170] Project website: https://wiki.digitalmethods.net/Dmi/DmiWinterSchool2012TrackingTheTrackers [Accessed 3 January 2013].

[171] In the case study I made use of Ghostery version 2.8.0.1 which contained a library of almost 1500 trackers. See: https://addons.mozilla.org/en-US/firefox/addon/ghostery/versions/?page=2#version-2.8.0.1 [Accessed 3 January 2013].

[172] See: https://addons.mozilla.org/en-us/firefox/addon/ghostery/versions/?page=1#version-2.5.3 [Accessed 3 January 2013].

[173] "Gephi is an interactive visualization and exploration platform for all kinds of networks and complex systems, dynamic and hierarchical graphs." See: https://gephi.org/ [Accessed 3 January 2013].

such as trackers, shows a specific view of the website's ecosystem. It allows for the reconstruction of a different network of connectivity operating in the back-end of a website showing the data flows between websites and tracking companies as will be demonstrated in the following case study.

Tracing the trackers using the Internet Archive Wayback Machine

In this case study I have developed a novel method to map the tracking ecology of the New York Times website over time by using the Tracker Tracker tool in combination with the Internet Archive Wayback Machine. I have made use of the Wayback Machine—the interface to the Internet Archive's web archives—because it provides a valuable source for web historians because of its accessibility and scope. The Wayback Machine contains over 435 Billion URLs[174] and provides snapshots from a wide range of archived websites from 1996 until very recent.

While trackers, or the websites issuing the trackers, may no longer exist or be in use, their code snippets—the code that enables the tracking—can still be found in archived snapshots of websites within the Wayback Machine. Trackers are visible in the archived website's source code because they are either "hardwired" in the source code or have become imprinted in the website during the archival process (see figure 24).

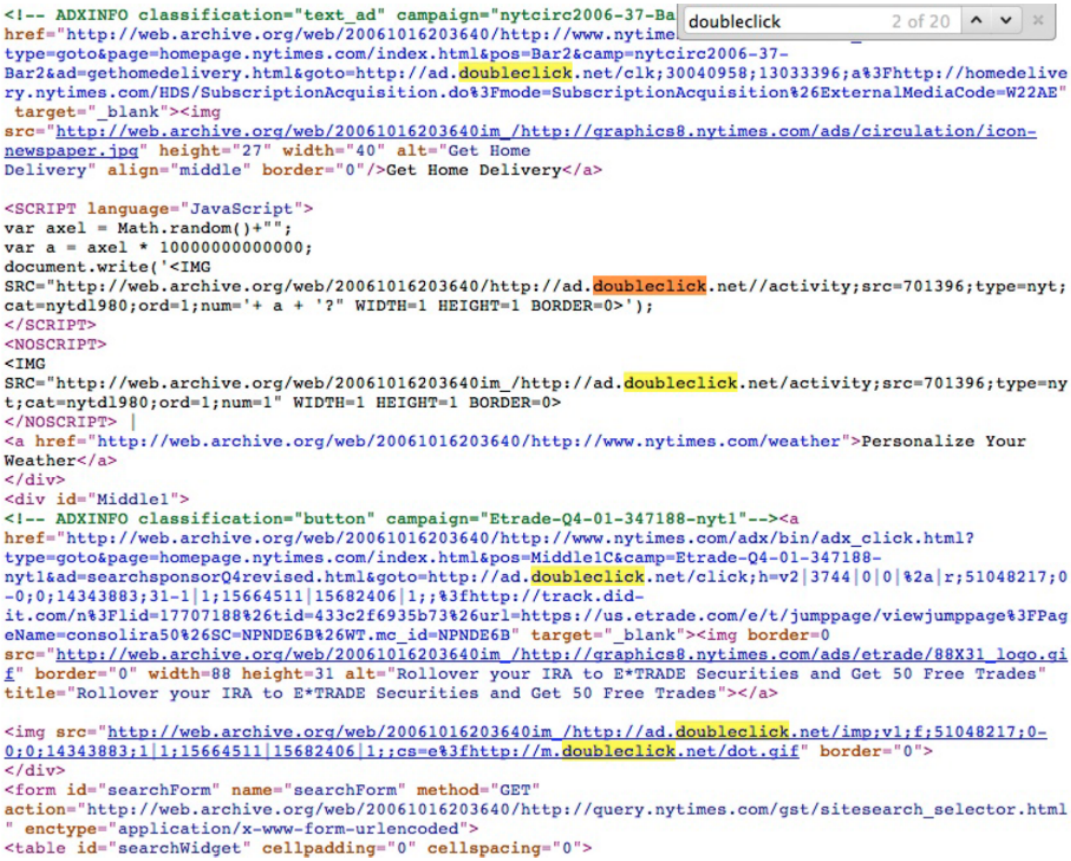
An important finding is that Ghostery still detects trackers in archived websites when surfing through the Wayback Machine with Ghostery enabled (see figure 25). When verifying the detected trackers, by manually comparing the source code of the archived website with the Ghostery database there is indeed a match with the pattern [ad.doubleclick.net]. Many patterns are established on such a level (for example domain name level [ad.doubleclick.net] or subpage level [doubleclick.net/pagead]) which means that if the tracking technique changes, the tracker will still be detected if it is issued from the same domain or subpage. According to Ghostery's developers trackers' issuing domains have hardly changed since they started developing the plugin, making it a suitable tool for historical research.[175] The question remains to what extent the Ghostery database contains old trackers. While the example of DoubleClick, which has existed since 1996, shows that such companies can be traced and detected in retrospect, the detection of old trackers relies on the assumption that tracker (sub)domains do not change over time. This means that the approach put forward here can only detect trackers that Ghostery currently has in its database and further research should therefore investigate old tracker patterns. Concerning new trackers, Ghostery operates a

[174] See: https://archive.org/web/ [Accessed 4 December 2014].

[175] On May 30, 2013 I was invited to visit Ghostery's office (Evidon Inc.) in New York City after they had learned about my research which I had presented earlier that month at the MIT7 conference in Cambridge, MA. During my brief visit I talked to Ghostery's developers and asked them about their tracker database and the plugin's technical functionality.

cumulative database and constantly adds new patterns of existing trackers to its database as well as new trackers.

What follows from the observation that Ghostery is still able to detect trackers in archived websites is first, that some of the functionality of the trackers continues to exist within archived websites,[176] and second, that the Tracker Tracker tool—which is based on Ghostery—also works with the Internet Archive's Wayback Machine. The latter makes it possible to detect and analyze a website's tracking ecology over time. Next, I will detail a method to do so.



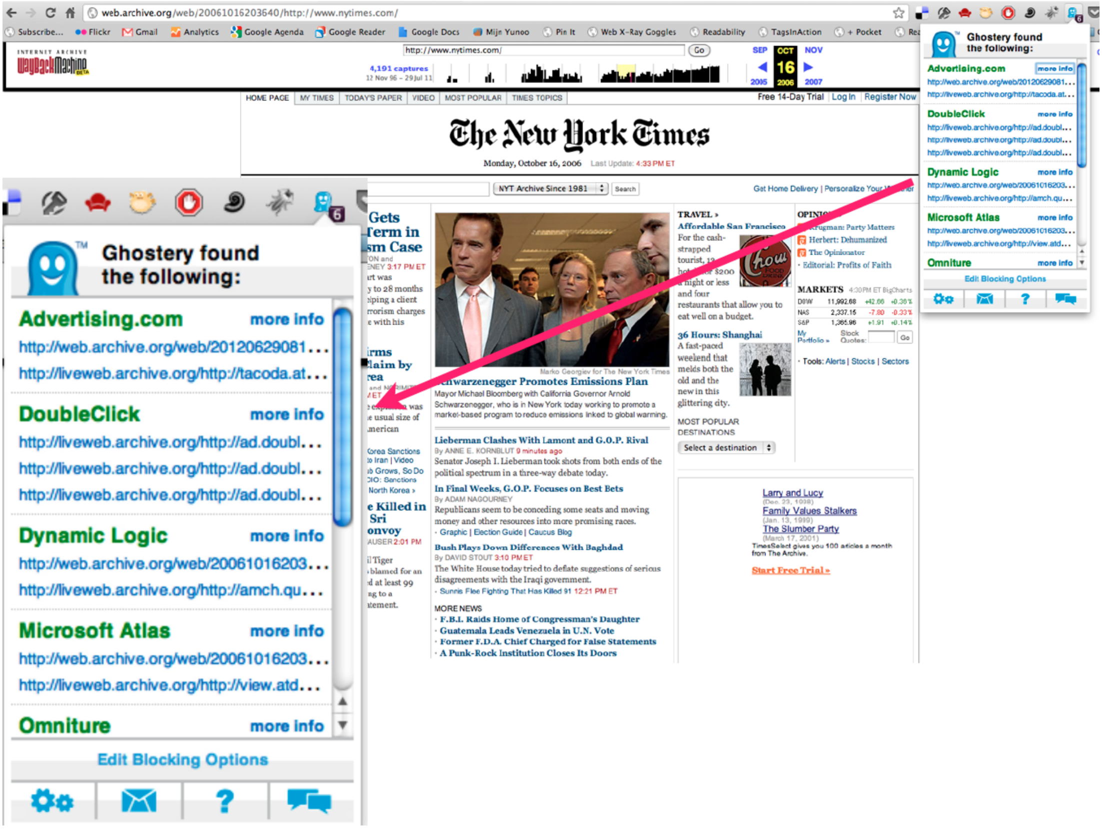
```
<!-- ADXINFO classification="text_ad" campaign="nytcirc2006-37-Ba
href="http://web.archive.org/web/20061016203640/http://www.nytime
type=goto&page=homepage.nytimes.com/index.html&pos=Bar2&camp=nytcirc2006-37-
Bar2&ad=gethomedelivery.html&goto=http://ad.doubleclick.net/clk;30040958;13033396;a%3Fhttp://homedelive
ry.nytimes.com/HDS/SubscriptionAcquisition.do%3Fmode=SubscriptionAcquisition%26ExternalMediaCode=W22AE"
 target="_blank"><img
src="http://web.archive.org/web/20061016203640im_/http://graphics8.nytimes.com/ads/circulation/icon-
newspaper.jpg" height="27" width="40" alt="Get Home
Delivery" align="middle" border="0"/>Get Home Delivery</a>

<SCRIPT language="JavaScript">
var axel = Math.random()+"";
var a = axel * 10000000000000;
document.write('<IMG
SRC="http://web.archive.org/web/20061016203640/http://ad.doubleclick.net//activity;src=701396;type=nyt;
cat=nytdl980;ord=1;num='+ a + '?" WIDTH=1 HEIGHT=1 BORDER=0>');
</SCRIPT>
<NOSCRIPT>
<IMG
SRC="http://web.archive.org/web/20061016203640im_/http://ad.doubleclick.net/activity;src=701396;type=ny
t;cat=nytdl980;ord=1;num=1" WIDTH=1 HEIGHT=1 BORDER=0>
</NOSCRIPT> |
<a href="http://web.archive.org/web/20061016203640/http://www.nytimes.com/weather">Personalize Your
Weather</a>
</div>
<div id="Middle1">
<!-- ADXINFO classification="button" campaign="Etrade-Q4-01-347188-nyt1"--><a
href="http://web.archive.org/web/20061016203640/http://www.nytimes.com/adx/bin/adx_click.html?
type=goto&page=homepage.nytimes.com/index.html&pos=Middle1C&camp=Etrade-Q4-01-347188-
nyt1&ad=searchsponsorQ4revised.html&goto=http://ad.doubleclick.net/click;h=v2|3744|0|0|%2a|r;51048217;0
-0;0;14343883;31-1|1;15664511|15682406|1;;%3fhttp://track.did-
it.com/n%3Flid=17707188%26tid=433c2f6935b73%26url=https://us.etrade.com/e/t/jumppage/viewjumppage%3FPag
eName=consolira50%26SC=NPNDE6B%26WT.mc_id=NPNDE6B" target="_blank"><img border=0
src="http://web.archive.org/web/20061016203640im_/http://graphics8.nytimes.com/ads/etrade/88X31_logo.gi
f" border="0" width=88 height=31 alt="Rollover your IRA to E*TRADE Securities and Get 50 Free Trades"
title="Rollover your IRA to E*TRADE Securities and Get 50 Free Trades"></a>

<img src="http://web.archive.org/web/20061016203640im_/http://ad.doubleclick.net/imp;v1;f;51048217;0-
0;0;14343883;1|1;15664511|15682406|1;;cs=e%3fhttp://m.doubleclick.net/dot.gif" border="0">
</div>
<form id="searchForm" name="searchForm" method="GET"
action="http://web.archive.org/web/20061016203640/http://query.nytimes.com/gst/sitesearch_selector.html
" enctype="application/x-www-form-urlencoded">
<table id="searchWidget" cellpadding="0" cellspacing="0">
```

Figure 24: Traces of a tracker in the source code of an archived front page of the New York Times in the Internet Archive Wayback Machine. The code snippets reveal the presence of DoubleClick ads and a DoubleClick tracking pixel [doubleclick.net/dot.gif]. The archived snapshot from the Wayback Machine is dated October 16, 2006. http://web.archive.org/web/20061016203640/http://www.nytimes.com [Accessed 12 January 2013].

[176] That is, tracking companies continue to track users within the archived web.



Ghostery found the following:

Advertising.com more info
http://web.archive.org/web/20120629081...
http://liveweb.archive.org/http://tacoda.at...

DoubleClick more info
http://liveweb.archive.org/http://ad.doubl...
http://liveweb.archive.org/http://ad.doubl...
http://liveweb.archive.org/http://ad.doubl...

Dynamic Logic more info
http://web.archive.org/web/20061016203...
http://liveweb.archive.org/http://amch.qu...

Microsoft Atlas more info
http://web.archive.org/web/20061016203...
http://liveweb.archive.org/http://view.atd...

Omniture more info

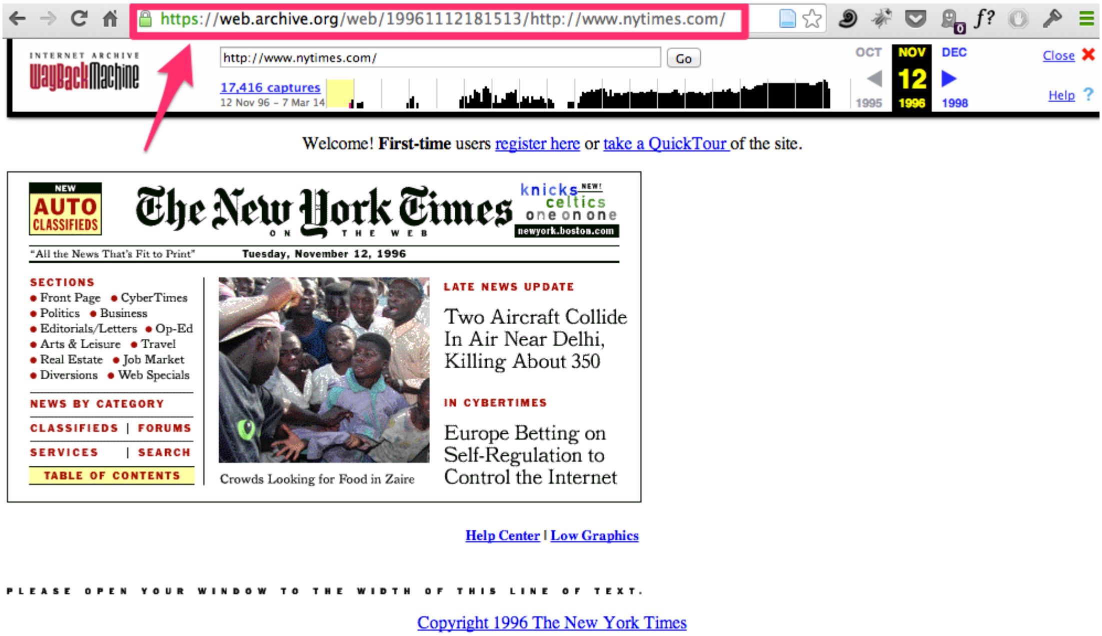
Edit Blocking Options

Figure 25: Ghostery detecting trackers on the archived front page of the New York Times website while using the Internet Archive Wayback Machine. The archived snapshot from the Wayback Machine is dated October 16, 2006. http://web.archive.org/web/20061016203640/http://www.nytimes.com [Accessed 12 January 2013].

The object of study is the archived New York Times website—or more specifically the archived New York Times front pages. The New York Times was chosen due to the site's centrality as a news source, its large number of visitors per day[177] and the presence of a fair amount of trackers—9 different trackers were detected by Ghostery on the New York Times front page at the time the pilot study was conducted in June 2012.[178] The selected time frame is 1996-2011, from the first (1996) to the final (2011) full year the New York Times was archived in the Internet Archive Wayback Machine at the time of the case study in December 2012. In a first step I set out to collect all the Internet Archive URLs for the archived snapshots from the New York Times in the Wayback Machine (see figure 26).

[177] According to Alexa the site currently ranks as the #114 within the top 1000 global websites and the #6 within the category of news websites. See http://www.alexa.com/siteinfo/nytimes.com and http://www.alexa.com/topsites/category/Top/News [accessed 12 January 2013].

[178] A pilot study to develop a methodology to analyze historical tracking ecologies was conducted with colleagues in June 2012, see: https://wiki.digitalmethods.net/Dmi/TracingTheTrackers [accessed 12 January 2013]. The case study presented here was done individually after I refined the methodology in December 2012.



https://web.archive.org/web/19961112181513/http://www.nytimes.com/

http://www.nytimes.com/

17,416 captures
12 Nov 96 - 7 Mar 14

Welcome! **First-time** users register here or take a QuickTour of the site.

NEW AUTO CLASSIFIEDS

The New York Times ON THE WEB

knicks NEW! celtics one on one newyork.boston.com

"All the News That's Fit to Print" **Tuesday, November 12, 1996**

SECTIONS

- Front Page
- CyberTimes
- Politics
- Business
- Editorials/Letters
- Op-Ed
- Arts & Leisure
- Travel
- Real Estate
- Job Market
- Diversions
- Web Specials

NEWS BY CATEGORY

CLASSIFIEDS | FORUMS

SERVICES | SEARCH

TABLE OF CONTENTS

Crowds Looking for Food in Zaire

LATE NEWS UPDATE

Two Aircraft Collide In Air Near Delhi, Killing About 350

IN CYBERTIMES

Europe Betting on Self-Regulation to Control the Internet

Help Center | Low Graphics

PLEASE OPEN YOUR WINDOW TO THE WIDTH OF THIS LINE OF TEXT.

Copyright 1996 The New York Times

Figure 26: The Internet Archive Wayback Machine URL for an archived snapshot of the New York Times website. The archived snapshot from the Wayback Machine is dated November 12, 1996. http://web.archive.org/web/20061016203640/http://www.nytimes.com [Accessed 12 January 2013].

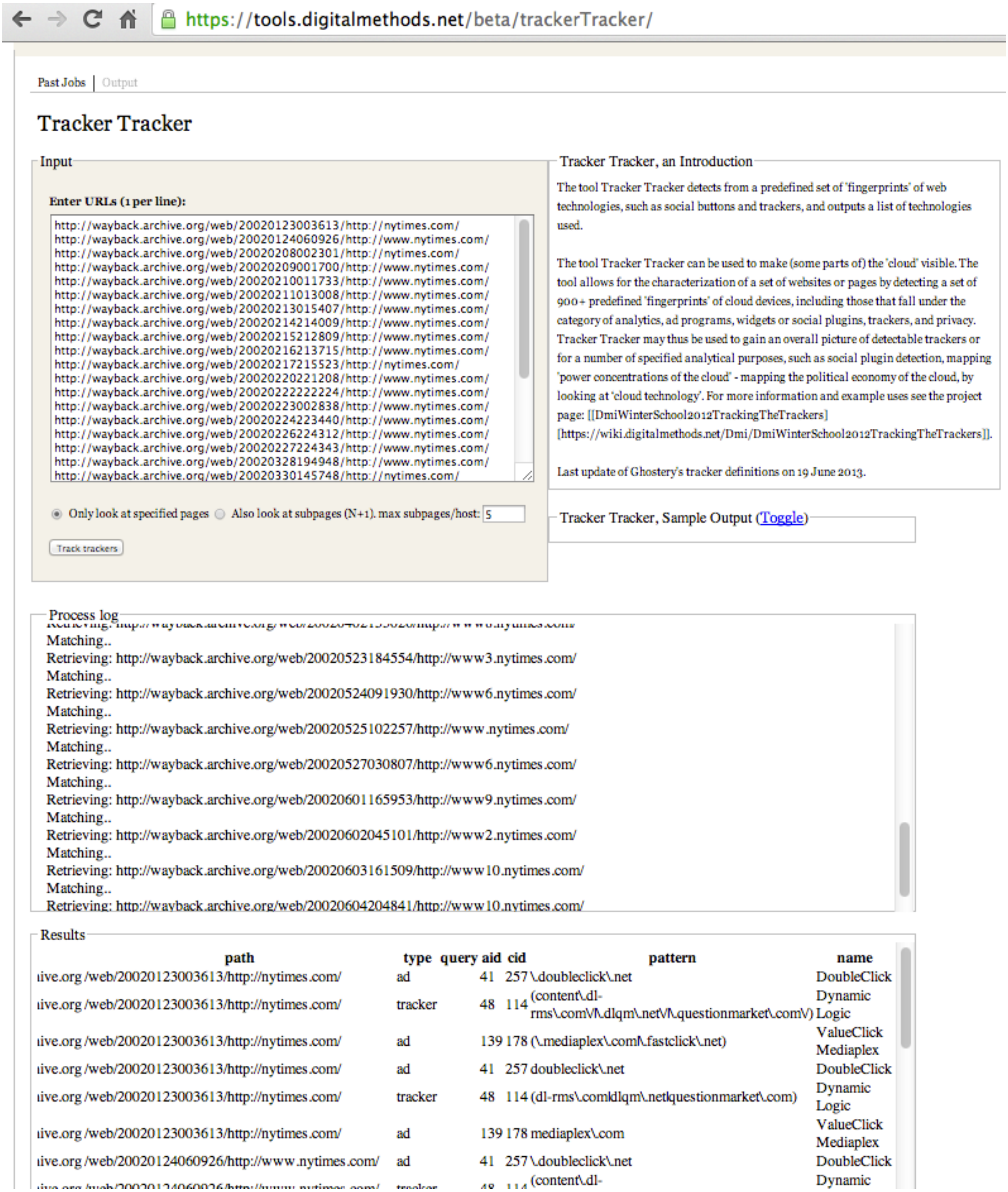
Instead of collecting all Internet Archive URLs manually, I have used the Internet Archive Wayback Machine Link Ripper tool to automate the process.[179] This tool retrieves the links of a website's archived snapshots at wayback.archive.org. The input is a URL [http://www.nytimes.com/] and the output is a text file which lists the Internet Archive URLs—the links of the archived snapshots. In case the Wayback Machine has archived multiple snapshots of the website per day, only the first archived version of that day is retained and listed in the resulting text file. Table 3 shows the number of URLs collected by the Internet Archive Wayback Machine Link Ripper per year.

'96	'97	'98	'99	'00	'01	'02	'03	'04	'05	'06	'07	'08	'09	'10	'11
7	4	1	13	10	174	157	71	21	230	208	290	261	243	281	352

Table 3: The number of Internet Archive Wayback Machine URLs collected for the New York Times website between 1996 and 2011 per year. The URLs were retrieved by the Internet Archive Wayback Machine Link Ripper tool.

[179] See: https://tools.digitalmethods.net/beta/internetArchiveWaybackMachineLinkRipper/ [accessed 12 January 2013].

In a second step I used the Tracker Tracker tool to scan the Internet Archive URLs—the archived snapshots—for tracking technologies. The input is the list of Internet Archive URLs that was compiled in the previous step and the output shows the detected trackers per URL (see figure 27).



https://tools.digitalmethods.net/beta/trackerTracker/

Past Jobs | Output

Tracker Tracker

Input

Enter URLs (1 per line):

```
http://wayback.archive.org/web/20020123003613/http://nytimes.com/
http://wayback.archive.org/web/20020124060926/http://www.nytimes.com/
http://wayback.archive.org/web/20020208002301/http://nytimes.com/
http://wayback.archive.org/web/20020209001700/http://www.nytimes.com/
http://wayback.archive.org/web/20020210011733/http://www.nytimes.com/
http://wayback.archive.org/web/20020211013008/http://www.nytimes.com/
http://wayback.archive.org/web/20020213015407/http://www.nytimes.com/
http://wayback.archive.org/web/20020214214009/http://www.nytimes.com/
http://wayback.archive.org/web/20020215212809/http://www.nytimes.com/
http://wayback.archive.org/web/20020216213715/http://www.nytimes.com/
http://wayback.archive.org/web/20020217215523/http://nytimes.com/
http://wayback.archive.org/web/20020220221208/http://www.nytimes.com/
http://wayback.archive.org/web/20020222222224/http://www.nytimes.com/
http://wayback.archive.org/web/20020223002838/http://www.nytimes.com/
http://wayback.archive.org/web/20020224223440/http://www.nytimes.com/
http://wayback.archive.org/web/20020226224312/http://www.nytimes.com/
http://wayback.archive.org/web/20020227224343/http://www.nytimes.com/
http://wayback.archive.org/web/20020328194948/http://www.nytimes.com/
http://wayback.archive.org/web/20020330145748/http://nytimes.com/
```

● Only look at specified pages ○ Also look at subpages (N+1). max subpages/host: 5

Track trackers

Tracker Tracker, an Introduction

The tool Tracker Tracker detects from a predefined set of 'fingerprints' of web technologies, such as social buttons and trackers, and outputs a list of technologies used.

The tool Tracker Tracker can be used to make (some parts of) the 'cloud' visible. The tool allows for the characterization of a set of websites or pages by detecting a set of 900+ predefined 'fingerprints' of cloud devices, including those that fall under the category of analytics, ad programs, widgets or social plugins, trackers, and privacy. Tracker Tracker may thus be used to gain an overall picture of detectable trackers or for a number of specified analytical purposes, such as social plugin detection, mapping 'power concentrations of the cloud' - mapping the political economy of the cloud, by looking at 'cloud technology'. For more information and example uses see the project page: [[DmiWinterSchool2012TrackingTheTrackers][https://wiki.digitalmethods.net/Dmi/DmiWinterSchool2012TrackingTheTrackers]].

Last update of Ghostery's tracker definitions on 19 June 2013.

Tracker Tracker, Sample Output (Toggle)

Process log

```
Retrieving: http://wayback.archive.org/web/20020402135020/http://www6.nytimes.com/
Matching..
Retrieving: http://wayback.archive.org/web/20020523184554/http://www3.nytimes.com/
Matching..
Retrieving: http://wayback.archive.org/web/20020524091930/http://www6.nytimes.com/
Matching..
Retrieving: http://wayback.archive.org/web/20020525102257/http://www.nytimes.com/
Matching..
Retrieving: http://wayback.archive.org/web/20020527030807/http://www6.nytimes.com/
Matching..
Retrieving: http://wayback.archive.org/web/20020601165953/http://www9.nytimes.com/
Matching..
Retrieving: http://wayback.archive.org/web/20020602045101/http://www2.nytimes.com/
Matching..
Retrieving: http://wayback.archive.org/web/20020603161509/http://www10.nytimes.com/
Matching..
Retrieving: http://wayback.archive.org/web/20020604204841/http://www10.nytimes.com/
```

Results

path	type	query	aid	cid	pattern	name
iive.org /web/20020123003613/http://nytimes.com/	ad		41	257	\.doubleclick\.net	DoubleClick
iive.org /web/20020123003613/http://nytimes.com/	tracker		48	114	(content\.dl-rms\.com\/\.dlqm\.net\/\.questionmarket\.com\/)	Dynamic Logic
iive.org /web/20020123003613/http://nytimes.com/	ad		139	178	(\.mediaplex\.com\.fastclick\.net)	ValueClick Mediaplex
iive.org /web/20020123003613/http://nytimes.com/	ad		41	257	doubleclick\.net	DoubleClick
iive.org /web/20020123003613/http://nytimes.com/	tracker		48	114	(dl-rms\.com|dlqm\.net|questionmarket\.com)	Dynamic Logic
iive.org /web/20020123003613/http://nytimes.com/	ad		139	178	mediaplex\.com	ValueClick Mediaplex
iive.org /web/20020124060926/http://www.nytimes.com/	ad		41	257	\.doubleclick\.net	DoubleClick
iive.org /web/20020124060926/http://www.nytimes.com/	tracker		48	114	(content\.dl-	Dynamic

Figure 27: The Tracker Tracker tool scanning all the Internet Archive Wayback Machine URLs for the New York Times websites between 1996 and 2011. The results show the name of the tracker, the type of tracker and the tracker pattern that were detected in the archived snapshots.

The result file can be downloaded from the tool in CSV (spreadsheet), GEFX (Gephi) or HTML-format and contains the Internet Archive URLs, the name of the tracker, the type of tracker and the tracker pattern that was detected. The type of trackers follows the categorization provided by Ghostery:[180]

> **Ad:** a tracker that delivers advertisements
> **Analytics**: a tracker that provides research/analytics for website publishers
> **Beacon/Tracker**: a tracker that exists only to track user behavior[181]
> **Widget**: a tracker that provides some kind of page function (comment forms, "Like" buttons, ...)

I then collected this detailed information in a spreadsheet. Figure 28 shows the number and type of trackers that have been detected in the archived snapshots of the New York Times front page per year.[182]

No trackers have been detected between 1996 and 2000. As discussed previously, ad servers and trackers have been an integral part of the web since the mid 1990s, so I manually verified the data of this period by checking the source code for the presence of third-party trackers. In 1996 and 1997 the front page of the New York Times is a clickable image map (see figure 26) and does not contain any trackers. In 1998 no trackers have been detected but manual verification revealed RealMedia ads (now 24/7 Media). In 1998 the New York Times had installed ad management platform Real Media on its own domain.[183] However, these ads are issued from the New York Times domain—making it a first-party tracker. Ghostery and the Tracker Tracker tool do not identify first-party trackers but focus on third-party trackers, trackers that are issued from a different domain (Knowlton 2010). In this case study I am interested in these third-party trackers because they indicate a relation with an external party.

As of 2001 the New York Times has started using a number of third-party advertising services: DoubleClick, LinkShare and Microsoft Atlas. The number of trackers increases per year and in 2006 and 2007 the New York Times front page contains 18 unique trackers over that year.[184] After 2007 there is a decline in the number of unique trackers which reflects the findings of the previously mentioned longitudal tracker study by Krishnamurthy and Wills who found an "increasing aggregation of user-related data by a steadily decreasing number of entities" (2009, 541). Further research could address this phenomenon by looking into whether this indicates media concentration in the ad network industry.

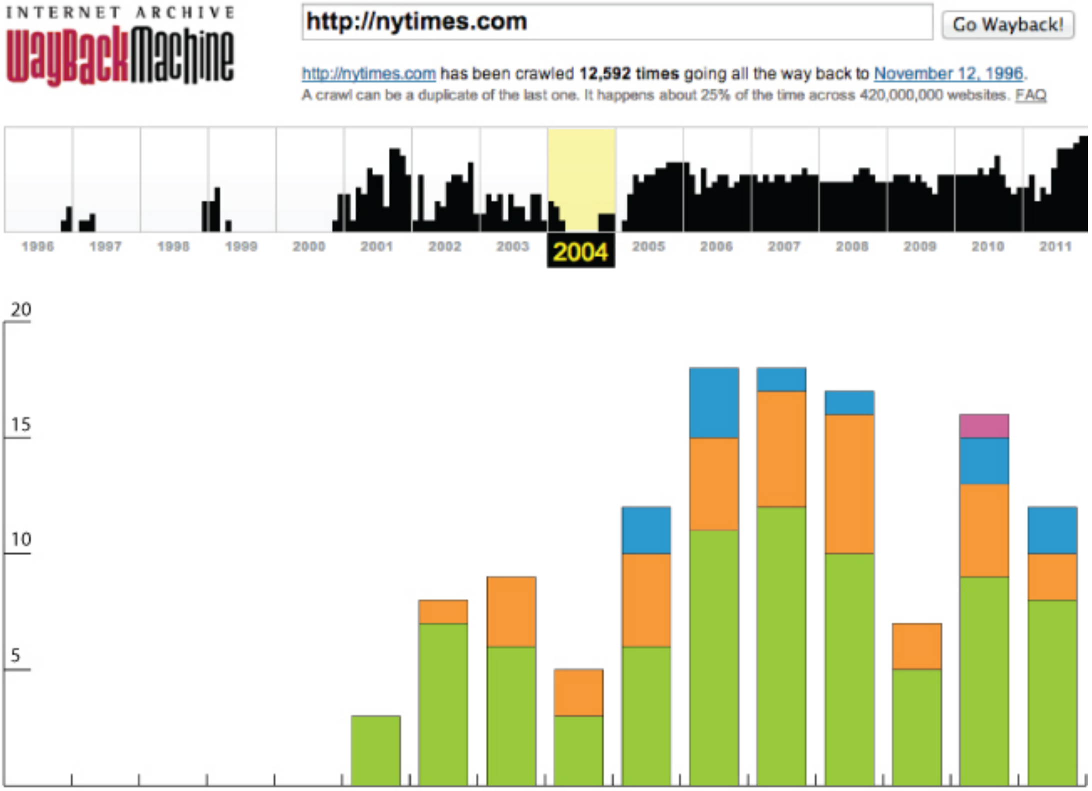
[180] See: http://www.knowyourelements.com/#tab=intro [accessed 12 January 2013].

[181] Ghostery has renamed the tracker type 'tracker' into 'beacon' since this case study has been conducted. I continue use the category name 'tracker' in this chapter to maintain consistency with Ghostery's naming of the tracker types at the time of the case study.

[182] There are a number of gaps in the available Internet Archive data from 1996-2000 and in 2004 data is missing for the months April-September which may explain the sudden "decline" in trackers in 2004.

[183] See: http://www.colepapers.net/TCP.archive/Cole_Papers_98/TCP_98_01/realmedia.HTML [Accessed 12 January 2013].

[184] The number of unique trackers detected per year does not mean that they were all present at the same time but that the NYT front page contained 18 unique trackers during the whole year.



http://nytimes.com has been crawled **12,592 times** going all the way back to November 12, 1996.
A crawl can be a duplicate of the last one. It happens about 25% of the time across 420,000,000 websites. FAQ

Figure 28: The number of trackers that have been detected by the Tracker Tracker tool per year on the archived New York Times front pages in the Internet Archive Wayback Machine between 1996 and 2011. The trackers have been color-coded according to the tracker types provided by Ghostery.[185] Green: ad, orange: tracker, blue: analytics, pink: widget. Visualization created with Adobe Illustrator. 2012.

Trackers in the form of widgets, which include social plugins, are relatively absent from the results. This can be explained by the setup of the research design. In this study I focused on the front page of the New York Times, whilst widgets such as social buttons are often not implemented on the front page but on the single-article page to like, share and tweet an article. Therefore I have requested an adjustment to the Tracker Tracker tool so it is now possible to scan up to five pages deep, starting from the homepage. This adjustment allows for tracing the rise of social plugins within archived websites over time for future research.

Figure 29 shows the diverse tracker environment the New York Times website has been embedded in over the years.

[185] See: http://www.knowyourelements.com/#tab=intro [accessed 12 January 2013].

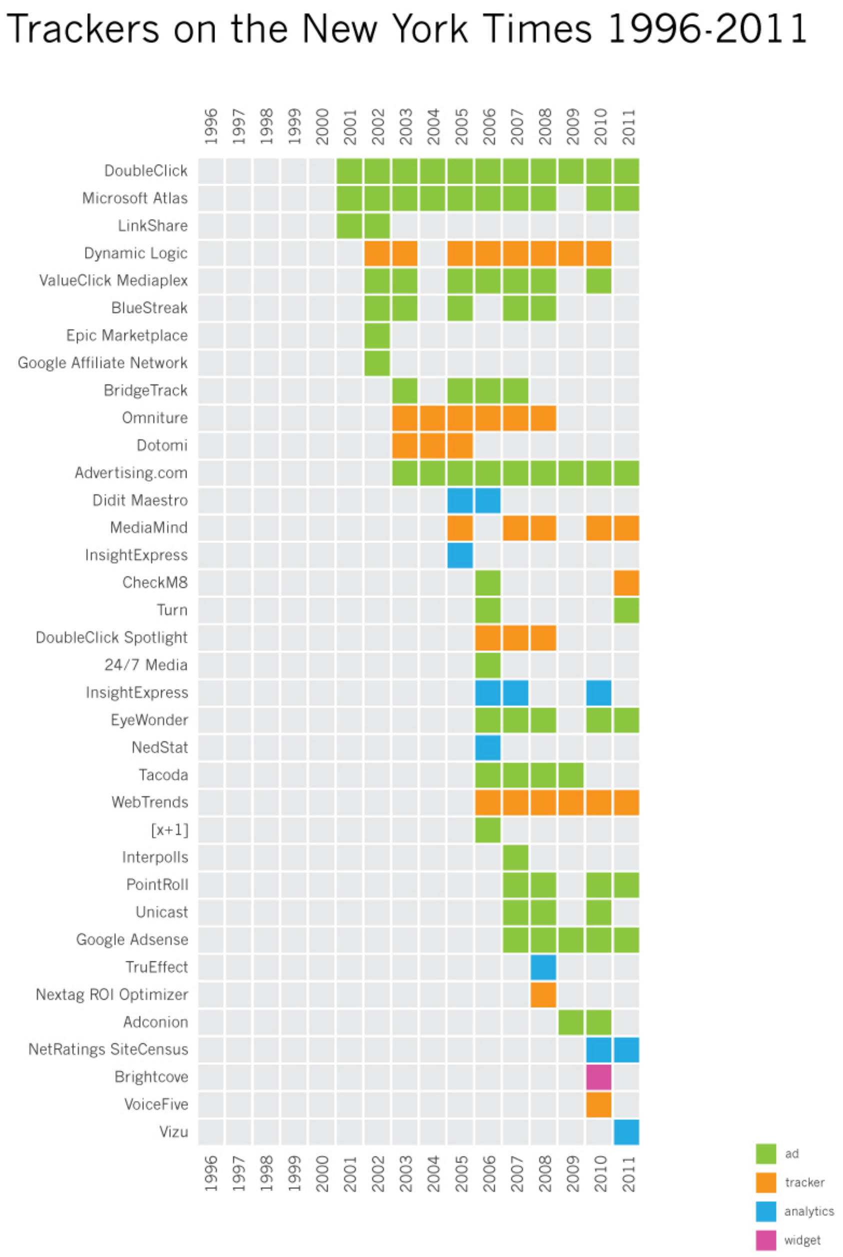


Figure 29: Names and types of the trackers that have been detected by the Tracker Tracker tool per year on the archived New York Times front pages in the Internet Archive Wayback Machine between 1996 and 2011. The trackers have been color-coded according to the tracker types provided by Ghostery.[186] Green: ad, orange: tracker, blue: analytics, pink: widget. Visualization created with Adobe Illustrator. 2012.

[186] See: http://www.knowyourelements.com/#tab=intro [accessed 12 January 2013].

This case study on the New York Times has demonstrated how we can employ historical source code analysis to analyze the historical tracker environment of a website using the Internet Archive's Wayback Machine. As such it has put forward a way in which an individual archived website can be used to uncover the interactions between the website and its environment.

One of the limitations of the case study is that the pattern of a specific tracker may change over time by using different scripts or tracking techniques (Orr et al. 2012). However, this case study has shown that in many cases patterns are established on domain name level and do not change significantly over time, e.g. [ad.doubleclick.com]. However, further research could include searching for possible databases that contain historical patterns in order to retrospectively update the pattern list.

In addition, in the future I wish to use the method presented in this chapter to scan a larger set of websites and move beyond the front page of these websites to detect social buttons. This builds on the research presented in chapter 5 in which I mapped the tracking ecology of the top 1000 Alexa websites (see figure 21). Scanning a large collection of archived websites over the timespan of 10 years would allow me to reconstruct and analyze the changing techno-commercial configurations of the web. This may be used to address the following questions: Who are the most prominent trackers over time? Are there any changes in the types of trackers (ads, analytics, widgets) over time? Is there a decline in the number of trackers used and does this point to media concentration? Microsoft Atlas for example—a prominent actor on the New York Times website in this study, was acquired by Facebook in February 2013.

This case study has contributed to web historiography as it shows how existing web archives can be used to study a website's ecosystem over time in order to analyze historical states of the web. Whilst this first case study has focused on employing code snippets to reconstruct a website's environment, the following case study puts forward a method to use the code snippets of social plugins to reconstruct missing social media content within archived websites.

The reconstructive affordances of APIs for web historians

In this second case study I return to the notion of the website as an assembled object from dynamic third-party sources (Mayer and Mitchell 2012). The dynamic programming language JavaScript has made it possible to load embedded resources within external websites, including tracking scripts and social plugins such as the Facebook Like Button or Facebook Comments plugin. However, these dynamic website elements pose a big challenge to traditional web archiving practices (Brügger 2013, 758) as employed by the Internet Archive Wayback Machine. In their Frequently Asked Questions the Internet Archive reports on the problems of archiving pages that are assembled from external and dynamic third-party content and functionality: "When a dynamic page contains forms, JavaScript, or other elements that require interaction with the originating host, the archive will not contain the original site's functionality" ("FAQ" 2014). As JavaScript is increasingly used to load content and functionality from external sources this has an enormous impact on the archivability of the web, Brunelle et al. contend (2015).

This problem should be framed within larger issues that web archives are facing in trying to keep up with the technological developments of the web (Jeffrey 2012; Ben-David and Huurdeman 2014). Current web archiving techniques are better tailored to websites as self-contained sources than websites as assembled from third-party elements (Ben-David and Huurdeman 2014).[187] I explore these issues by looking at an archived website, the news website Huffington Post, which makes use of a number of dynamic third-party objects including Facebook's Social Plugins. In particular, I look at a single article of the Huffington Post on the topic of ISIS, which has generated over 400 comments on its page through a third-party commenting system provided by Facebook.[188]

Webmasters can implement the Facebook Comments plugin to create Facebook-enabled comment spaces on their pages (see figure 30).[189] On the live web[190] these Facebook Comments are loaded from Facebook's platform in real-time (see figure 28).[191]

[187] It is beyond the scope of this paper to go into the problems of archiving social media platforms, see the work of McCown and Nelson (2009), SalahEldeen and Nelson (2012) and Jeffrey (2012).

[188] URL of the Huffington Post article: http://www.huffingtonpost.com/azeem-ibrahim/by-destroying-churches-an_b_6927242.html. URL of the Huffington Post article in the Wayback Machine: https://web.archive.org/web/20150413155142/http://www.huffingtonpost.com/azeem-ibrahim/by-destroying-churches-an_b_6927242.html [Accessed 12 April 2015].

[189] "The Comments Plugin lets people comment on content on your site using their Facebook account. If people wish to they can share this activity to their friends in News Feed as well. It also contains built-in moderation tools and special social relevance ranking." See: https://developers.facebook.com/docs/plugins/comments/ [Accessed 12 April 2015].

[190] The term "live web" follows the terminology used by the Internet Archive and Niels Brügger to refer to "what is on the web today" (2009, 125).

[191] "The Comments Plugin lets people comment on content on your site using their Facebook account. If people wish to they can share this activity to their friends in News Feed as well. It also contains built-in moderation tools and special social relevance ranking." See: https://developers.facebook.com/docs/plugins/comments/ [Accessed 12 April 2015].

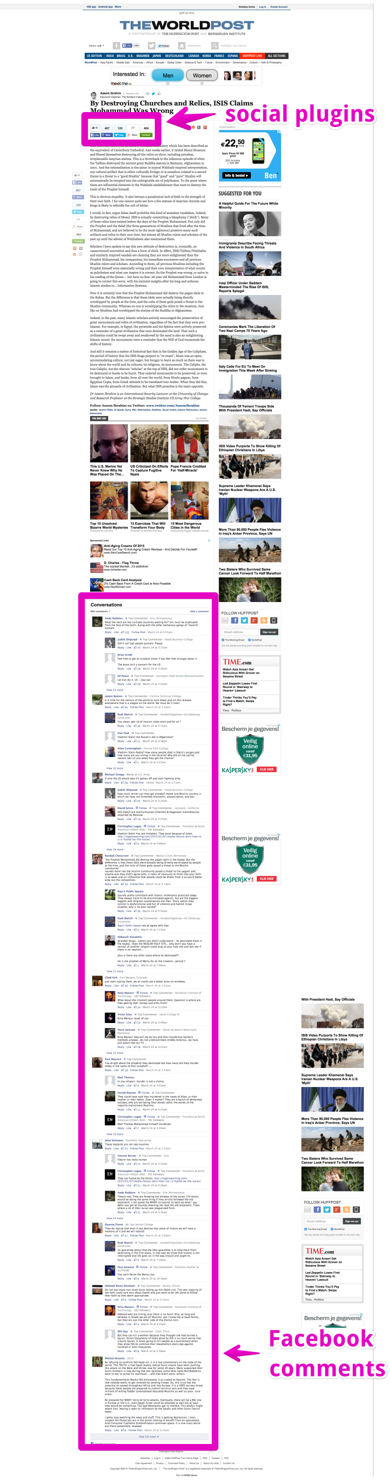
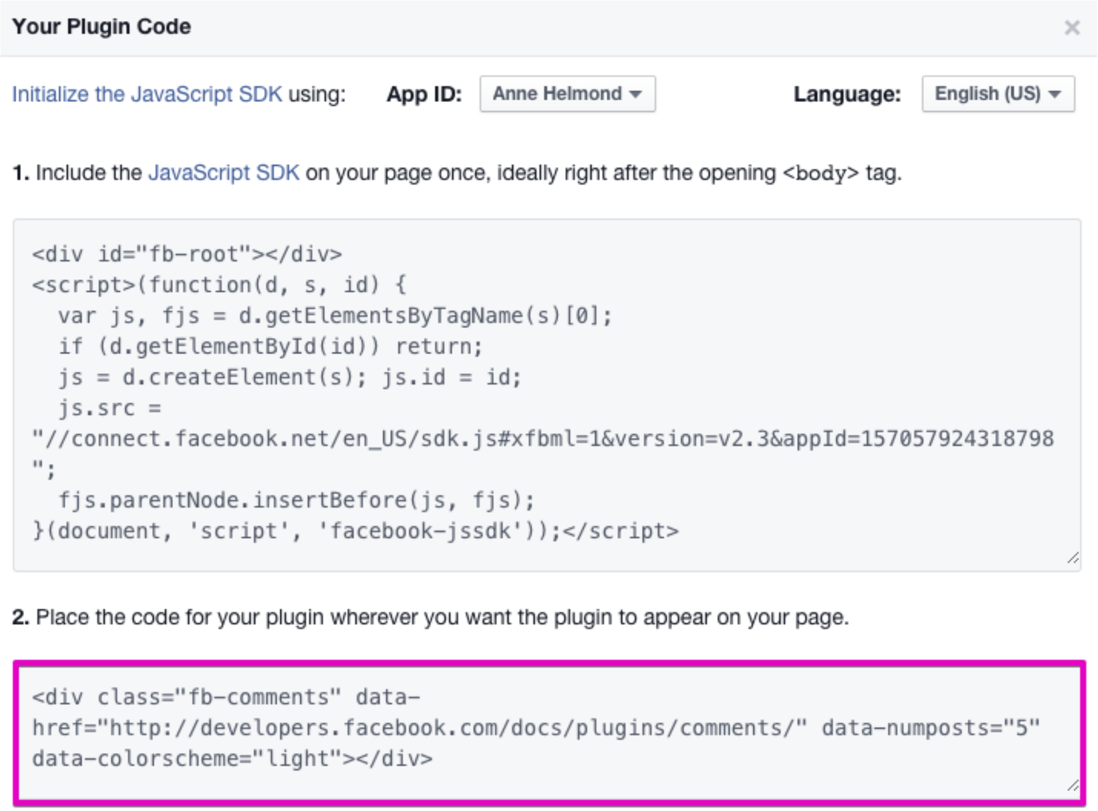


Figure 30: Live web view of the Huffington Post article on ISIS from March 23, 2015: http://www.huffingtonpost.com/azeem-ibrahim/by-destroying-churches-an_b_6927242.html [Accessed 12 April 2015]. The presence of social plugins and Facebook comments on the page has been highlighted.

The Facebook Comments plugin can be integrated into a website by including two pieces of code: First, Facebook's JavaScript SDK, and second, the plugin (see figure 31).



Your Plugin Code

Initialize the JavaScript SDK using: App ID: Anne Helmond Language: English (US)

1. Include the JavaScript SDK on your page once, ideally right after the opening <body> tag.

```
<div id="fb-root"></div>
<script>(function(d, s, id) {
  var js, fjs = d.getElementsByTagName(s)[0];
  if (d.getElementById(id)) return;
  js = d.createElement(s); js.id = id;
  js.src =
"//connect.facebook.net/en_US/sdk.js#xfbml=1&version=v2.3&appId=157057924318798
";
  fjs.parentNode.insertBefore(js, fjs);
}(document, 'script', 'facebook-jssdk'));</script>
```

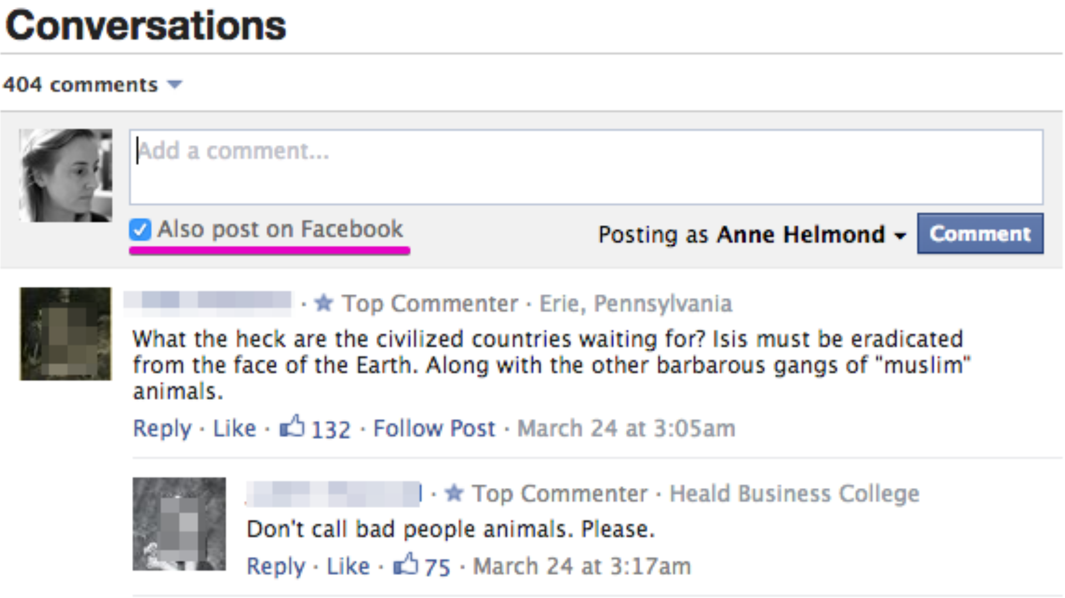
2. Place the code for your plugin wherever you want the plugin to appear on your page.

```
<div class="fb-comments" data-
href="http://developers.facebook.com/docs/plugins/comments/" data-numposts="5"
data-colorscheme="light"></div>
```

Figure 31: The plugin code for the Facebook Comments plugin. The plugin code consists of two sets of code snippets: 1. The JavaScript SDK establishes a communication channel between the website and Facebook, and 2. The plugin code creates a data pour on the website for Facebook comments. Retrieved from: https://developers.facebook.com/docs/plugins/comments/ [Accessed 12 April 2015].

The plugin sets up a data channel by establishing a connection with Facebook's database to send and retrieve comments to and from the platform. Technically, the plugin functions as an API call into Facebook's database (see chapter 3), meaning it sends specific operations to Facebook to perform tasks such as: get user profile, get total number of comments, get five comments on this article, write new comment, like comment, etc. This data is then rendered and displayed within the plugin on the website.

As addressed in chapter 2, the plugin is what Alan Liu refers to as a "data pour," a snippet of code on a website that creates a container for sending and retrieving content from and to an external database (2004, 59). When a user leaves a comment using Facebook Comments this comment is not only posted on the website but also—as enabled through the default settings—posted to the user's Newsfeed in Facebook (see figure 32).



Conversations

404 comments ▾

Add a comment...

☑ Also post on Facebook

Posting as Anne Helmond ▾ Comment

[redacted] · ★ Top Commenter · Erie, Pennsylvania

What the heck are the civilized countries waiting for? Isis must be eradicated from the face of the Earth. Along with the other barbarous gangs of "muslim" animals.

Reply · Like · 👍 132 · Follow Post · March 24 at 3:05am

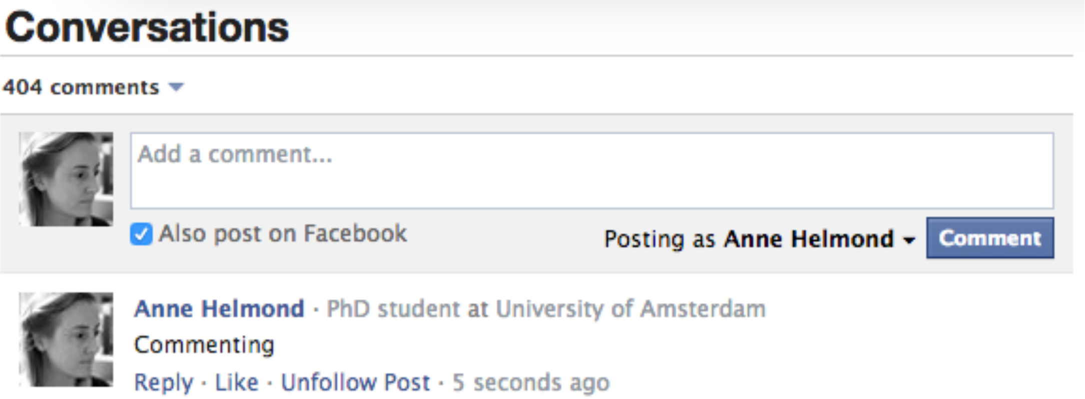
[redacted] · ★ Top Commenter · Heald Business College

Don't call bad people animals. Please.

Reply · Like · 👍 75 · March 24 at 3:17am

Figure 32: The Facebook Comments plugin as shown on the Huffington Post website. The functionality to cross-post the comment to Facebook has been highlighted. Screenshot from the Facebook comment space on the Huffington Post website: http://www.huffingtonpost.com/azeem-ibrahim/by-destroying-churches-an_b_6927242.html [Accessed 1 May 2015].

Similar to the functionality of the Like button (see chapter 5), the comment can gather additional comments on the website through replies (see figure 32 and 33) and within Facebook's Newsfeed through comments (see figure 34), scaling up future engagement.



Conversations

404 comments ▾

Add a comment...

☑ Also post on Facebook

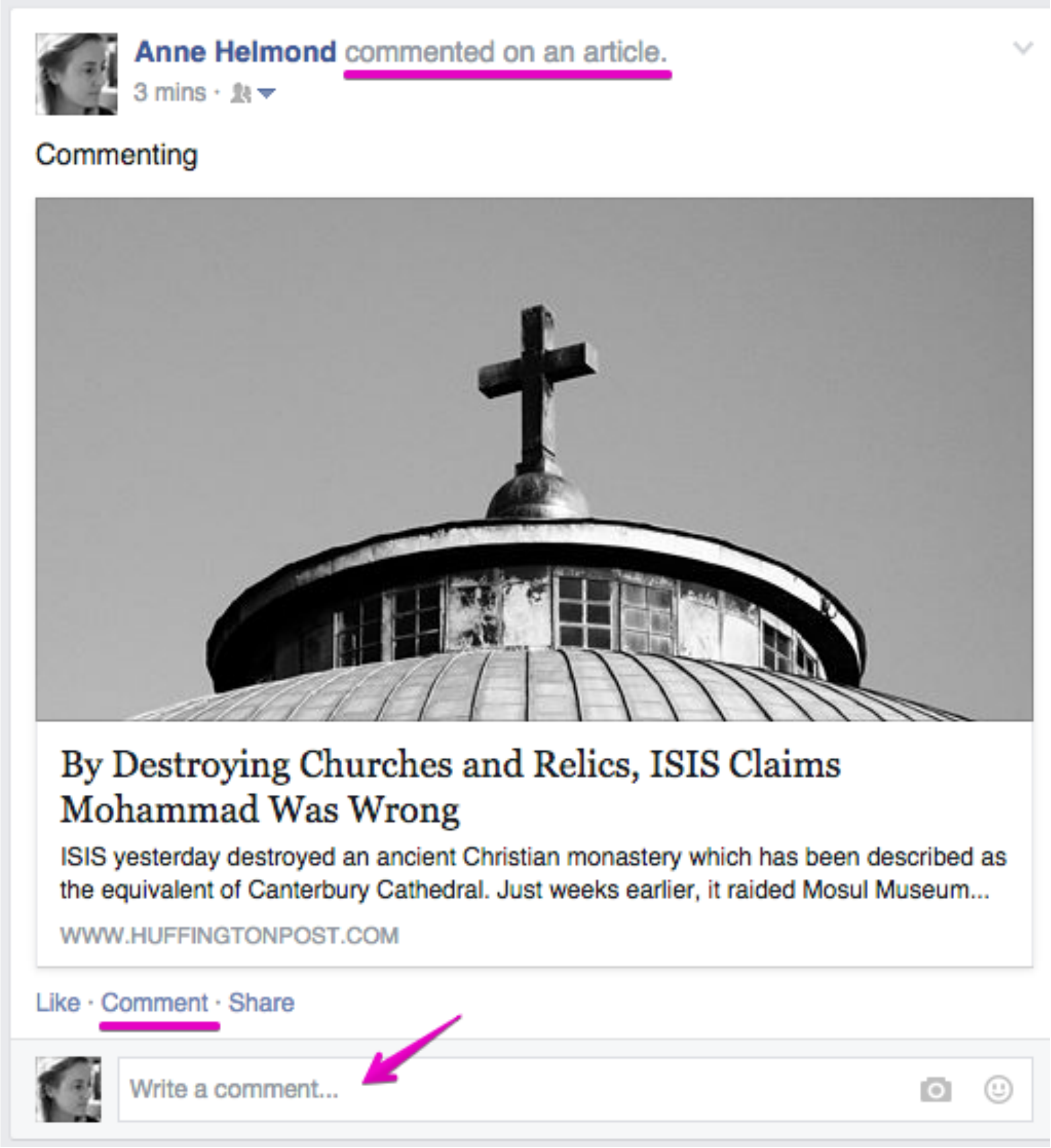
Posting as Anne Helmond ▾ Comment

Anne Helmond · PhD student at University of Amsterdam

Commenting

Reply · Like · Unfollow Post · 5 seconds ago

Figure 33: The Facebook Comments plugin as shown on the Huffington Post website. The screenshot shows the posted comment on the Huffington Post. Screenshot from the Facebook comment space on the Huffington Post website: http://www.huffingtonpost.com/azeem-ibrahim/by-destroying-churches-an_b_6927242.html [Accessed 1 May 2015].



Anne Helmond commented on an article.

3 mins ·

Commenting

By Destroying Churches and Relics, ISIS Claims Mohammad Was Wrong

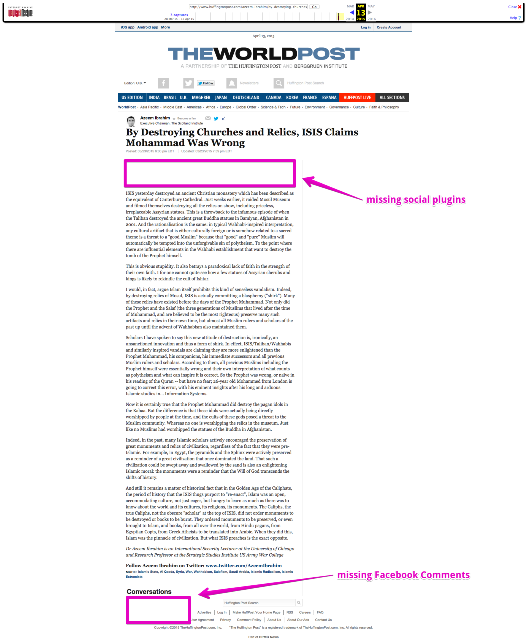
ISIS yesterday destroyed an ancient Christian monastery which has been described as the equivalent of Canterbury Cathedral. Just weeks earlier, it raided Mosul Museum...

WWW.HUFFINGTONPOST.COM

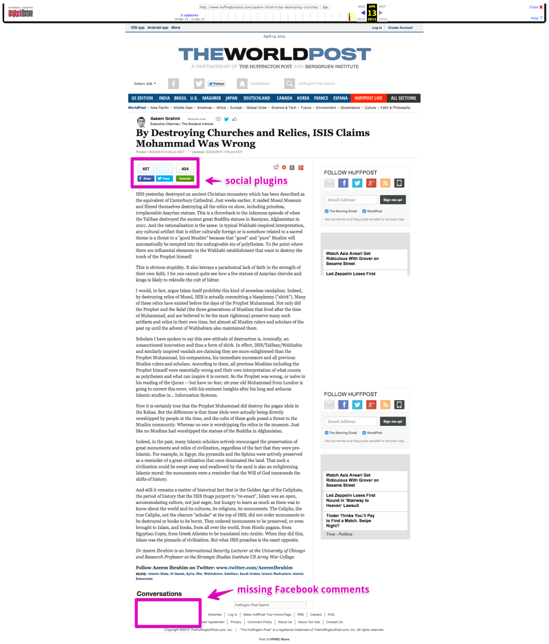
Figure 34: The cross-posted comment on the Facebook Newsfeed. The screenshot shows the result of cross-posting a comment from the Huffington Post website to Facebook using the Facebook Comments plugin. Facebook users can further comment on the Huffington Post article on the Facebook Newsfeed. Screenshot from the author's Facebook News Feed.

Comments produced on the website and within Facebook are synchronized across both spaces. This turns the data channel of the plugin into a lively infrastructure (Beer 2013; Marres and Weltevrede 2013) as data produced in one space immediately informs the other and is continuously updated (see chapter 5). The plugin, or data pour, acts as an exchange mechanism for data, comments, which are not part of the web page itself but which are requested from Facebook's database when the page and Comments plugin are loaded.

Considering the impact of JavaScript on archivability (Brunelle et al. 2015) and the Internet Archive's own cautious remarks about dynamically generated webpages, I turn to the Wayback Machine to see how the Huffington Post article has been archived. As web archiving theorist Niels Brügger has argued in detail, the archived website is not an exact copy of the one on the live web but a *unique version* as the result of the archival process (Brügger 2008, 156). Archived websites, Megan Sapnar Ankerson adds, often suffer from "temporal and technical inconsistencies" (2012, 387). I noticed such irregularities when I received at least two different renderings of the archived Huffington Post article within a time span of five minutes: In figure 35 the social plugins as well as the Facebook comments are missing and in figure 36 the social plugins have been rendered but not the Facebook comments.


THE WORLDPOST

By Destroying Churches and Relics, ISIS Claims Mohammad Was Wrong


THE WORLDPOST

By Destroying Churches and Relics, ISIS Claims Mohammad Was Wrong

Figure 35 (left): The archived snapshot of the Huffington Post article on ISIS from March 23, 2015. URL of the Huffington Post article in the Wayback Machine. Retrieved from: https://web.archive.org/web/20150413155142/http://www.huffingtonpost.com/azeem-ibrahim/by-destroying-churches-an_b_6927242.html [Accessed 12 April 2015]. In the screenshot, view A (left), the missing social plugins and missing Facebook Comments in the archived snapshot have been highlighted.

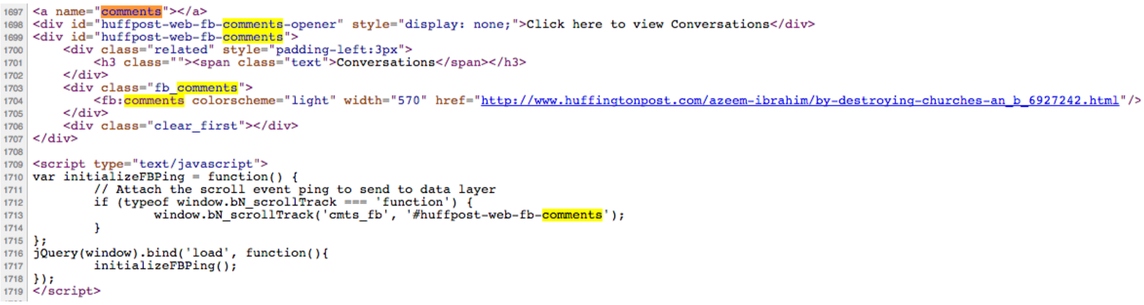
Figure 36 (right): The archived snapshot of the Huffington Post article on ISIS from March 23, 2015. URL of the Huffington Post article in the Wayback Machine. Retrieved from: https://web.archive.org/web/20150413155142/http://www.huffingtonpost.com/azeem-ibrahim/by-destroying-churches-an_b_6927242.html [Accessed 12 April 2015]. In the screenshot, view B (right), the social plugins and missing Facebook Comments in the archived snapshot have been highlighted.

This volatile rendering of the same archived website poses a challenge for the web historian who wishes to examine these features or their content. How do we know a comment space existed on this website if it is not rendered in the archived version or if it has been detached from the archived website in the archival process?

To answer these questions, I move from the user interface to the archived source code as suggested by Niels Brügger: "All in all, with regard to deficient link structures and missing elements or functions, the overall method is: click on anything, use the source code, and examine every corner of the archived website, even if it appears useless at first glance" (2008, 166). While embedded content and functionality may not be archived or rendered, the code that embeds or refers to these objects, the data pour, is imprinted in the source code. The archived source code provides the entry point for the discovery and subsequent reconstruction of missing content. As explained previously, Facebook provides webmasters with a code snippet to integrate the Comments plugin into their websites:

```
<div class="fb-comments" data-href="http://developers.facebook.com/docs/plugins/comments/"
data-numposts="5" data-colorscheme="light"></div>.[192]
```

This makes it fairly easy to locate the plugin in the source code by searching for code snippets such as [fb-comments] or [comments] (see figure 37). Now that I have located the presence of the Facebook Comments Plugin in the archived source code, I move on to the question of how we can retrieve the missing comments from Facebook's database.



```
<a name="comments"></a>
<div id="huffpost-web-fb-comments-opener" style="display: none;">Click here to view Conversations</div>
<div id="huffpost-web-fb-comments">
    <div class="related" style="padding-left:3px">
        <h3 class=""><span class="text">Conversations</span></h3>
    </div>
    <div class="fb_comments">
        <fb:comments colorscheme="light" width="570" href="http://www.huffingtonpost.com/azeem-ibrahim/by-destroying-churches-an_b_6927242.html"/>
    </div>
    <div class="clear_first"></div>
</div>

<script type="text/javascript">
var initializeFBPing = function() {
        // Attach the scroll event ping to send to data layer
        if (typeof window.bN_scrollTrack === 'function') {
                window.bN_scrollTrack('cmts_fb', '#huffpost-web-fb-comments');
        }
};
jQuery(window).bind('load', function(){
        initializeFBPing();
});
</script>
```

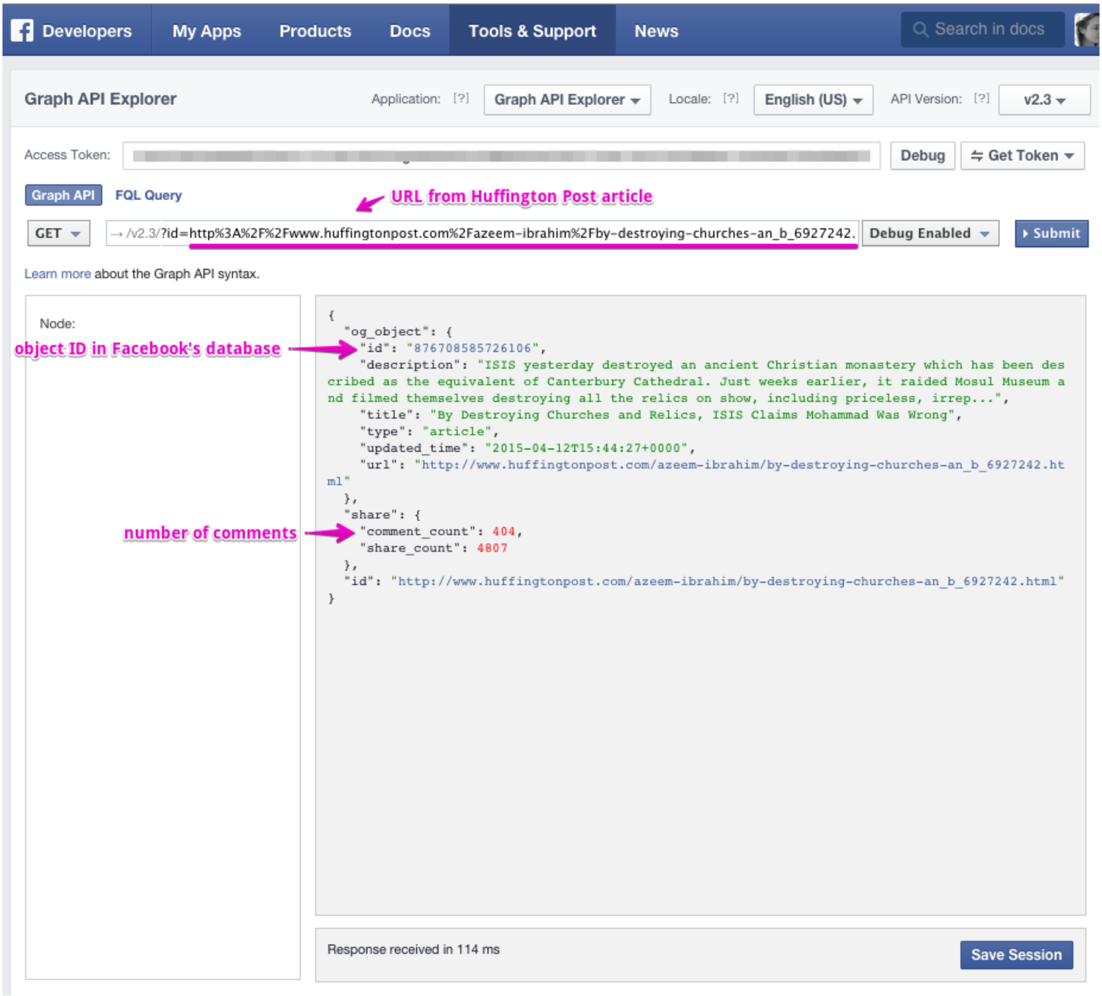
Figure 37: The source code of the archived Huffington Post article in the Wayback Machine. Retrieved from: https://web.archive.org/web/20150413155142/http://www.huffingtonpost.com/azeem-ibrahim/by-destroying-churches-an_b_6927242.html [Accessed 12 April 2015]. In the screenshot the code snippets that point to the presence of the embedded Facebook Comments plugin have been highlighted.

To recover the comments, I turn to Facebook's Graph API as "the primary way to get data in and out of Facebook's platform."[193] First, I need to locate the Huffington Post article in Facebook's database. Every item, such as an external news page with Facebook Comments,

[192] See: https://developers.facebook.com/docs/plugins/comments [Accessed 27 April 2015].

[193] See: https://developers.facebook.com/docs/graph-api/overview/ [Accessed 27 April 2015].

receives a unique ID to identify the object within the Facebook platform. Using the Graph API Explorer[194] we can request the ID for a specific URL, in this case the Huffington Post article (see figure 38).[195]



URL from Huffington Post article

→ /v2.3/?id=http%3A%2F%2Fwww.huffingtonpost.com%2Fazeem-ibrahim%2Fby-destroying-churches-an_b_6927242.

object ID in Facebook's database

number of comments

```
{
  "og_object": {
    "id": "876708585726106",
    "description": "ISIS yesterday destroyed an ancient Christian monastery which has been des
cribed as the equivalent of Canterbury Cathedral. Just weeks earlier, it raided Mosul Museum a
nd filmed themselves destroying all the relics on show, including priceless, irrep...",
    "title": "By Destroying Churches and Relics, ISIS Claims Mohammad Was Wrong",
    "type": "article",
    "updated_time": "2015-04-12T15:44:27+0000",
    "url": "http://www.huffingtonpost.com/azeem-ibrahim/by-destroying-churches-an_b_6927242.ht
ml"
  },
  "share": {
    "comment_count": 404,
    "share_count": 4807
  },
  "id": "http://www.huffingtonpost.com/azeem-ibrahim/by-destroying-churches-an_b_6927242.html"
}
```

Response received in 114 ms

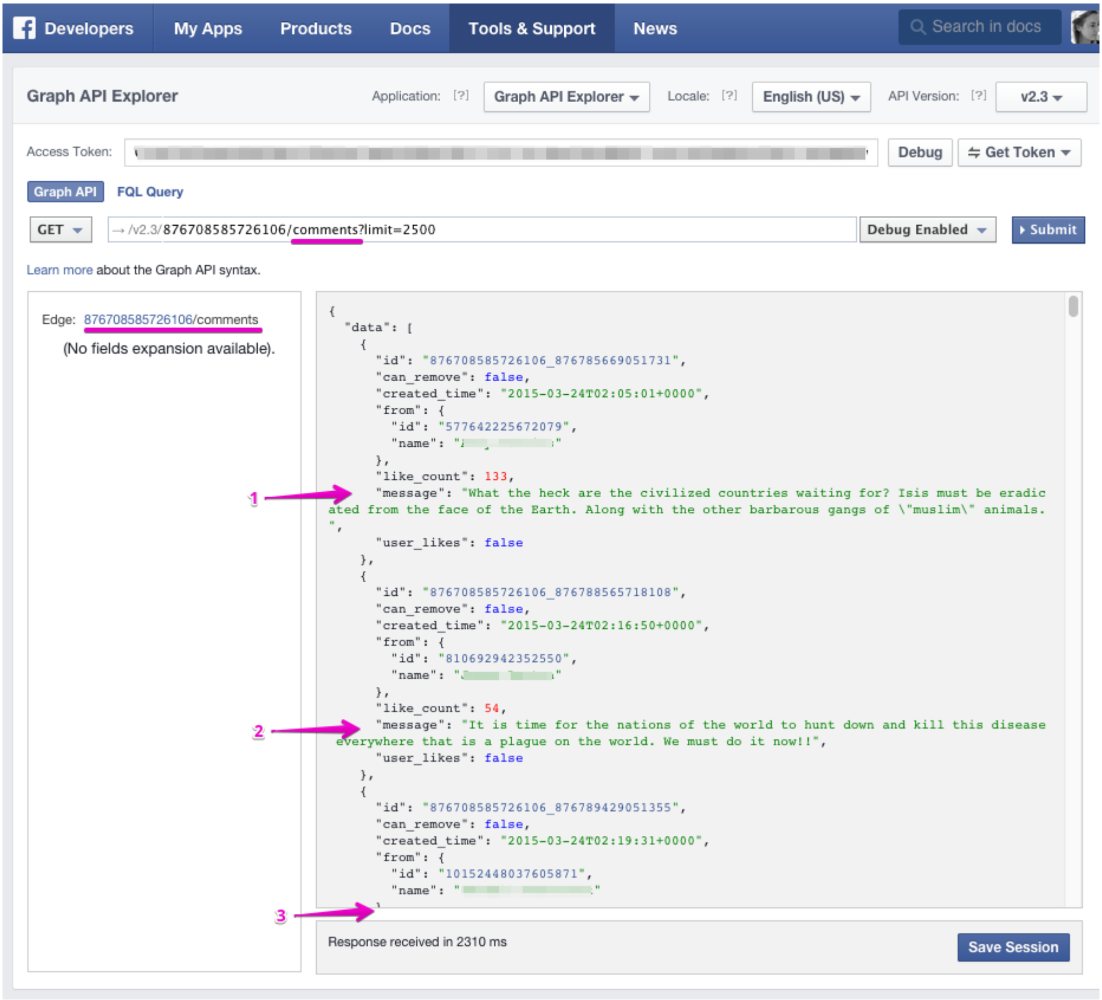
Figure 38: The Facebook Graph API Explorer displaying the Facebook ID for a URL. Retrieved from: https://developers.facebook.com/tools/explorer/ [Accessed 27 April 2015]. In the screenshot the input [an encoded URL] and output [available data about this URL within Facebook's platform] of the Graph API Explorer have been highlighted.

Now that we have the object ID, "876708585726106,"[196] we can request the comments related to this object (see figure 39).[197] The Graph API explorer displays all comments, including the

[194] See: https://developers.facebook.com/tools/explorer/ [Accessed 27 April 2015].

[195] See: https://developers.facebook.com/docs/graph-api/reference/v2.3/url [Accessed 27 April 2015].

time the comment was created, the name of the commenter (anonymized in figure 39) and the number of likes a comment has received. The data is provided in JSON, a data format that can be read by humans and processed by machines for further analysis. A web archivist could write a script calling the Facebook Graph API to request all comments for a set of archived websites under analysis.[198]



Graph API Explorer

Application: [?] Graph API Explorer ▾ Locale: [?] English (US) ▾ API Version: [?] v2.3 ▾

Access Token: [redacted]

GET ▾ → /v2.3/876708585726106/comments?limit=2500 Debug Enabled ▾

Learn more about the Graph API syntax.

Edge: 876708585726106/comments

(No fields expansion available).

```
{
  "data": [
    {
      "id": "876708585726106_876785669051731",
      "can_remove": false,
      "created_time": "2015-03-24T02:05:01+0000",
      "from": {
        "id": "577642225672079",
        "name": "[redacted]"
      },
      "like_count": 133,
      "message": "What the heck are the civilized countries waiting for? Isis must be eradic
ated from the face of the Earth. Along with the other barbarous gangs of \"muslim\" animals.
",
      "user_likes": false
    },
    {
      "id": "876708585726106_876788565718108",
      "can_remove": false,
      "created_time": "2015-03-24T02:16:50+0000",
      "from": {
        "id": "810692942352550",
        "name": "[redacted]"
      },
      "like_count": 54,
      "message": "It is time for the nations of the world to hunt down and kill this disease
 everywhere that is a plague on the world. We must do it now!!",
      "user_likes": false
    },
    {
      "id": "876708585726106_876789429051355",
      "can_remove": false,
      "created_time": "2015-03-24T02:19:31+0000",
      "from": {
        "id": "10152448037605871",
        "name": "[redacted]"
      }
```

Response received in 2310 ms

Figure 39: The Facebook Graph API Explorer displaying the comments for the Facebook ID. Retrieved from: https://developers.facebook.com/tools/explorer/ [Accessed 27 April 2015]. In the screenshot the input [request comments for ID] and output [available comment data] of the Graph API Explorer have been highlighted.

[196] Basic information about the object ID can be retrieved from: https://graph.facebook.com/876708585726106 [Accessed 27 April 2015].

[197] See: https://developers.facebook.com/docs/graph-api/reference/v2.3/comment [Accessed 27 April 2015].

[198] Comments can also be read through the Facebook Graph API by following the instructions provided in this chapter.

This second case study has demonstrated that APIs cannot only be used as tools for data collection for the live web (Lomborg and Bechmann 2014), but also for the archived web. It could be argued that social media platforms have intrinsic archives since, Robert Gehl contends, they have been built to capture, store, process and organize massive amounts of user data in structured databases (Gehl 2011, 1232). However, Gehl explains, "just data sets are not in themselves archives. To be an archive, the material collected must be done in an organized manner that allows for the *post hoc* construction of power" that is "[t]he material collected must be done in anticipation of its future reconstruction." Here, I see Facebook as an archive[199] for two reasons: First, Facebook employs its Open Graph infrastructure (see chapter 2 and 5) to collect, process and organize data in a very structured manner (Bucher 2012a, 5). Second, I argue that by exposing data to third parties through the Graph API, Facebook invites developers to build new applications on top of Facebook's data, which I see as a form of anticipatory reconstruction.

In addition, APIs can be also be considered as "data makers" (Vis 2013) that can enrich the archived snapshot of a website through the reconstruction of social media platform data. They can further enrich archived websites by giving access to additional data, such as comments that have been posted after the website has been archived, or the number of likes and shares a website currently has. As such, APIs and their associated social plugins, operate as a lively infrastructure (Beer 2013; Marres and Weltevrede 2013), as a software interface between web archives and the real-time social web.

The use of social media APIs for web archeological purposes does come with a number of considerations because APIs function as protocological objects (Bucher 2013) and regulatory instruments (Puschmann and Burgess 2013) that carefully regulate access to data. Access can be shut down or limited at any moment, creating a volatile arrangement between web archeologists, those trying to dig up content, and social media platforms.

Web histories: Reconstructing past states of the web using code snippets

This chapter has aimed to contribute to the growing field of web historiography by putting the Internet Archive's Wayback Machine to new uses. One of the questions within this field is what kinds of web histories can be told using web archives. Dominant approaches focus on the history of a single site or a network of sites through historical hyperlink analysis because of the Internet Archive Wayback Machine's focus on the unit of the website at the expense of the website's larger context (Brügger 2013; Rogers 2013; Ben-David and Huurdeman 2014). The Internet Archive Wayback Machine does not only detach an archived website from its context but also from its dynamic content. Within the web as platform, the website is increasingly

[199] See also Beer (2013), Kaun and Stiernstedt (2014) and Hogan (2015) on the notion of Facebook as an archive.

assembled from dynamic third-party sources, posing a challenge to current archival techniques. Often, the content of these dynamic objects is missing in the website's archived snapshot, or has been frozen in the archival process. In this chapter I have shown how the code snippets of third-party objects in the archived source code can be used to address these two problems that web historians are facing: missing content and missing context. The archived source code contains information about a website's relationships with third parties. In particular, the archived code snippets of third-party objects provide an entry point for reconstructing the website's historical ecosystem and its missing platform content.

Thus, I have proposed to reconceptualize the study of websites as website ecology. By developing a method which enables web historians to scan websites for third-party dynamic objects over time, I have proposed to add the study of historical website ecosystems to the field of web historiography. This case study has presented a methodological approach to examine the platformization of the web over time using web archives. In addition, I have explored the reconstructive affordances of APIs as valuable sources for web historians to reconstruct missing social media content in archived websites. I have developed a method to locate and retrieve missing social media platform content using archived source code and APIs.

7. Conclusion: Studying the platformization of the web and beyond

In this dissertation I have analyzed how social media platforms have transformed the web's infrastructure. I have argued that the rise of social media has introduced the platform as the dominant infrastructural and economic model of the social web. The mid 2000s saw the launch of social media sites such as Flickr, YouTube, Facebook and Twitter, which, in the years to follow, would all turn into platforms. I have located this moment of social network *sites* becoming social media *platforms* in their offering of an API. An Application Programming Interface (API) provides access to a site's data and functionality and turns the site into a programmable platform. As computational platforms these social network sites have made themselves accessible to third parties such as webmasters and app developers. In doing so, social media platforms have enabled third-party developers to build new applications on top of their platforms and to integrate platform features into their own websites and apps. I have put forward that this programmability is a key characteristic of social media platforms allowing for their dissemination. I have further demonstrated how they employ platform-specific objects such as social plugins to further extend themselves into the web by decentralizing their platform data and functionality whilst simultaneously recentralizing data produced on external websites back to the platforms' databases.

This integration of platform features into other parts of the web has created new types of relationships between platforms and websites in which data flows between users, websites, platforms and other parties such as ad servers. As an example of this redistribution, I have foregrounded social buttons, which communicate with a platform's database through API calls, as one of the prime devices to establish such data channels between social media platforms and third parties. I have argued that whilst platforms are establishing these relations they are simultaneously deploying their platform infrastructure to format external web data entering the platform to fit its underlying logic. This extension of social media platforms into the web and their use of the platform infrastructure to make external web data platform ready is what I have referred to as the double logic of the platformization. In this process, platforms are employing their APIs, which enact the platform's programmability, to connect their infrastructural model of decentralizing platform features to their economic model of recentralizing platform ready data.

One of the key arguments of this dissertation is that this particular use of platform model has enabled the platformization of the web—turning other web spaces into instantiations of social media—which has consequences for how the web is organized. In five case studies I have traced the rise of the platform model and have examined the effects of the platformization of the web on the web's infrastructure. I have sought to address the impact of social media on the web not by focusing on its use and its users, but rather, by zooming in on an understudied aspect of social media, the role of software platforms which power the social web. By turning to software studies and platform studies, I have foregrounded the function of the platform infrastructure in creating new types of relations between various actors on the web.

This dissertation is concerned with the way in which social media platforms have extended into the web, or as O'Reilly phrased it: "The Web 2.0 lesson: leverage customer-self service and algorithmic data management to reach out to the entire web, to the edges and not just the center, to the long tail and not just the head" (2005). In order to understand *what social media has done to the web*, I have focused on studying changes in the underlying infrastructure of the web over time in relation to the rise of social media platforms. This historical perspective on social media through the lens of platformization reflects on past states of the web not to be nostalgic about "the web we lost" (Dash 2012)—although we might have sufficient reasons to be—but rather to provide a detailed understanding of how the web that we currently inhabit has come into being through particular choices made. By looking at how the web's infrastructure was organized in the past and has been reconfigured by social media, we can begin to unfold and understand the complex social media ecology of the present.
In each chapter I have provided a medium-specific (Rogers 2013) history of the rise of social media and platform-specific methods to analyze the platformization of the web. Each case study has set out to contribute to the growing field of software studies and the related field of platform studies and their calls for new methodological development (Bogost and Montfort 2009; Langlois, McKelvey, et al. 2009; Beer 2009; Manovich 2013) by providing novel methods analyzing the role of the platform in the changing infrastructure of the web. Learning from existing software studies and platform politics approaches (Langlois, McKelvey, et al. 2009; Gillespie 2010; Bucher 2012b) I have developed new methods that I frame as digital methods for platform studies. Let me recapitulate the specific contributions made throughout the case studies.

In chapter 2 on 'The platformization of the web' I have historicized the notion of platformization by providing an overview of developments through which social network *sites* have become social media *platforms*. In particular, this chapter has focused on the coming into being of the material-technical infrastructures of social media platforms by providing a historical perspective on the technological development of software platforms on the web. I have done so by tracing the advent of social media APIs as part of the material infrastructure of social media platforms. Doing so, I have taken a medium-specific approach to platforms to trace the emergence of this specific architecture on the web and its consequences. With a focus on the role of the platform's defining characteristic, its programmability through the API and related set of social buttons, I have shown the work that social media platforms do in a computational sense (cf. Bogost and Montfort 2009; cf. Gillespie 2010). I have examined how websites have historically enacted their programmability before the widespread availability of APIs and how they have previously enabled data exchanges to extend beyond their boundaries. Employing the approach based on disaggregation (Langlois, McKelvey, et al. 2009) I have traced and analyzed distinct components of data exchange mechanisms in the social web. As a result, I have located three pre-conditions for the platformization of the web: the separation of content and presentation as illustrated with XML, the modularization of content and features through widgets, and the interfacing with databases through APIs. Together these three components reveal how the architectures of social media platforms are geared towards expanding into other online spaces and transforming external web data. To understand this process I have drawn on Alan Liu's idea of the "data pour" (2004), that is embedded code on a page that pulls in and displays dynamic content from third-party databases. The notion has allowed me to examine how social media platforms are setting up *two-way data channels* between their platform and external websites through widgets that directly interface with the platform's database. It is here that the politics of platforms, that is the work that platforms do not only rhetorically but also computationally (cf. Gillespie 2010), becomes visible. The data pours do not merely function as channels (cf. Gillespie 2010; cf. Van Dijck 2013c) through which data flows to and from the platform's database, but data flowing back into the platform is re-formatted according to the logic of the platform. I have demonstrated the changing politics of data flows in the social web where early platform widgets were devised to decentralize a platform's data and functionality, whilst later platform plugins have been developed to also recentralize data back to the platform and to employ the platform infrastructure to make this external web data platform ready.

Moving from the techno-material development of the platform model towards the integration of platform features into other parts of the web, in chapter 3 on 'The coming of the platforms' I have examined the role of social media in the Dutch blogosphere. I have turned to the blogosphere because bloggers have played an important role in weaving social media platforms into the rest of the web by embedding platform content into their blog posts and by using sidebar widgets. In this chapter I have contributed to personal accounts of the history and evolution of the Dutch blogosphere (Meeuwsen 2010) and historical blogosphere analysis (Ammann 2009; Stevenson 2010b) by mapping the rise and decline of the Dutch blogosphere. In addition I have empirically examined the symbiotic relationship between social media platforms and blogs in order to understand infrastructural changes in the Dutch blogosphere.

Bloggers make use of a number of blog-related services such as social media platforms to embed external functionality and content through widgets. In doing so, bloggers have pioneered a new linking practice in connecting websites not by conventional hyperlinks but rather through platform features such as widgets. These linking practices have transformed the link structure of the blogosphere and have called into question the traditional definition of the blogosphere as an interlinked network of blogs, rather they have embedded the blogosphere into the social media platform ecology. In order to analyze infrastructural changes in a blogosphere over time we first need to construct a representation of its hyperlink network. In this chapter I have contributed to the historical hyperlink analysis of blogs (Ammann 2009;

Stevenson 2010b) with a method that creates yearly snapshots of a blogosphere by employing the outlinks of archived blogs from the Internet Archive Wayback Machine. In addition, I have put forward a method to examine the relationship between blogs and social media platforms by distinguishing between different link types.

In chapter 4 on 'The algorithmization of the hyperlink' I focused on the "industrialization" (Turow 2008) of the hyperlink by social media platforms. I have drawn from the previous findings to further analyze how social media platforms have altered the link structure of the web by examining the changing role of the hyperlink itself. I have argued that we can analyze the impact of social media on the web's infrastructure by analyzing how social media platforms have transformed the main structural element of the web, the natively digital object of the hyperlink, into the platform-specific object of the shortened URL. Starting from the observation that social media platform Twitter automatically creates shortened URLs from links shared to its platform, I have turned to analyze the role of the platform infrastructure in this process. By unpacking how the software platform handles hyperlinks we can learn more about the role of hyperlinks within the medium and how this relates to the platform's politics (Gillespie 2010). Following this line of inquiry, I have traced the commodification of the hyperlink first by search engines that turned the link into the currency of the web and later by social media platforms that turned the link into an analytical device. In the social web, Langlois et al. have previously argued (2009), the production of links is no longer a manual task reserved for webmasters and bloggers, but instead performed by blog software and social media platforms. These platforms, I have argued, have created automated forms of linking producing links that reflect the underlying logic of the platform. These software-generated links in the social web are formatted to fit the purpose of the platform and to feed the underlying algorithms and analytics suites. I have foregrounded how social buttons act as important devices automatically creating data-rich shortened URLs for platforms to collect valuable data through link statistics by tracking users' interaction with links across the web. To demonstrate how social media platforms use their platform infrastructure to automatically create such data-rich links I have developed a novel method that traces the redirect path of a shortened URL. By following a shortened URL we can see how each platform reconfigures the hyperlink in order to make it platform ready and analyze the political economy of linking in the social web (cf. Walker 2002) by mapping the configuration of actors involved in the creation, proliferation and distribution of links. These actors, I have argued, are involved in building a data-rich infrastructure of platform-specific shortened URLs on top of the traditional long URLs.

In chapter 5 on 'The Like economy' I have further analyzed how webmasters, users and social media platforms are creating such data-intensive infrastructures through social buttons by focusing on the Facebook Like Button. I have done so by providing a historical perspective on the rise of social buttons as metrics of user engagement in relation to different web economies: the hit, link and Like economy. Within the informational web, or Web 1.0, the web-native objects of the hit and the link have been central to organizing economic value production online. Search engines turned the key web-native object of the hyperlink into the currency of the web (Rogers 2002; Walker 2002) by employing the links created between websites by webmasters to measure the relative importance of a website. Within the social web, or Web 2.0, social buttons have emerged as a new and distributed way to validate web content through web activities such as liking, sharing and tweeting. Social buttons have become important devices to turn this user engagement into numbers on button counters which function as comparable metrics within the emerging Like economy of the social web. In doing so social media platforms are turning user affect and engagement into data and objects of exchange. I have demonstrated how the Like economy of the social web extends the hit and link economies of the informational web by being able to connect the validation of content—in the form of web activities such as liking—to a user's social media platform profile. One of the key findings of this chapter is that social media platforms have shifted the currency of the web from web-native, that is hits and links, to platform-native, that is likes, shares and retweets. The creation of platform-specific currencies should be seen in relation to changing linking practices within the social web. Previously, in the informational web, hyperlinks were mainly manually created by webmasters and openly accessible to search engines that employed them to determine a site's value. Whereas these links served as the currency of the open web, likes can be seen as the currency of an increasingly proprietary social web. That is, social media platforms are creating currencies that are tied to the mechanics and logics of their own platform infrastructure.

In this chapter I have also demonstrated how Facebook's Like economy is enabled by the interconnected dynamics of the decentralization of data production—by offering social buttons to like content across the web—and the recentralization of data collection through these buttons. In doing so, Facebook is creating an alternative fabric of the web, which is not based on hyperlinks between websites but on data connections between websites and Facebook's platform databases. These new types of connections are enabled through social buttons and their associated tracking cookies that create data flows in the back-end of Facebook's platform. Whereas in the previous chapter I developed a method to map the shortened URL networks created by social buttons, in this chapter I have developed a novel method to map the spread of social buttons across websites to map the data-intensive infrastructures created by social media platforms. I have done so by repurposing the browser plugin Ghostery, which detects trackers—such as Facebook's Like button—on websites to detect these objects across a collection of websites. Thus, this method allows for mapping the decentralization of social buttons into the web and the recentralization of platform ready data. It allows for exploring the alternative fabric of web—which is co-created by social media platforms offering social plugins and by webmasters integrating those plugins—by mapping the tracking ecology related to a set of websites. That is, the method enables the mapping of the larger techno-commercial environments that websites are embedded in by tracing the data connections between websites and other actors on the web.

In the final chapter on 'Website ecologies' I have expanded this method to map the changing composition of the web over time from the perspective of the website. In this case study I have demonstrated how we can use the code snippets of third-party objects in websites, including social media platform buttons, to map the ecosystem of a website. The website and its archived source code have been positioned as the objects of study (cf. Schneider and Foot 2004; cf. Brügger 2009) through which we can analyze the web's infrastructural composition and the platformization of the web at large. Reconceptualizing the study of archived website's as historical website ecology has been positioned as a theoretical and methodological contribution to web historiography (Brügger 2009; Ankerson 2009).

In the early days of the web, websites were fairly self-contained units where content was stored on the same server (Song 2010). Whilst such sites still exist, in the social web, however, websites are increasingly shaped by third-party content and functionality (J. R. Mayer and Mitchell 2012). I have argued that the platformization of the web has made the boundaries of the website porous by creating data channels or data pours between websites and platforms. We can thus see the website as an ecosystem in which particular relations with other actors on the web become inscribed. This chapter has further demonstrated how the web does not only become interconnected through hyperlinks (Elmer and Langlois 2013) but also through users' web activities, software, platform features, and data flows. Here, I have traced what Elmer and Langlois refer to as new forms of "networked connectivity" that move beyond the hyperlink and have turned to their notion of cross-platform analysis to map the interrelations between websites and platforms through these networked connectivity objects (2013). In order to map the wider networked ecology that websites are embedded in, I have proposed a method that relies on the source code of websites to look for the code snippets of third-party objects. In a second step I have further adjusted this method so that we can analyze historical website ecologies over time by making use of archived websites from the Internet Archive's Wayback Machine.

Whilst the archived source code of websites enables us to reconstruct previous states of the web and trace the platformization of the web, I have also demonstrated that the code snippets of third-party objects such as social plugins can be employed to reconstruct missing social media content from archived websites. Current archival techniques, such as those employed by the Internet Archive, are not attuned to websites assembled from dynamic third-party content. That is, the data flowing through the data pours of social plugins are often frozen in the process of archiving, or not archived at all. However, the code snippets of these data pours provide a valuable resource to reconstruct this missing data. I have provided a method to use these code snippets in archived websites to request the content from Facebook's database by tracing the API calls in the social button's code. I have argued that APIs, and their associated set of social buttons, operate as a lively infrastructure by providing a software interface between web archives and the real-time social web.

Each of the chapters in this dissertation has addressed the main question of what social media has done to the web in different ways. One of the key findings is that social media platforms use their platform features to decentralize data production and recentralize it for further processing. In doing so, platforms are making data passing through their infrastructure platform ready. This data feeds into the algorithms that sort and rank content on the platform as well as into the analytical suits behind the platforms. These practices indicate that we are moving towards a web which is made social media ready. This raises important questions for further research, especially since social media platforms such as Twitter and Facebook are increasingly accessed through and interwoven with apps. First, let me summarize the main contributions of this dissertation for the emerging fields of software studies, platform studies and digital methods.

Key contributions

In this dissertation I have developed a platform critique that revolves around the notion of platformization and its consequences. Platformization can be understood as the effects of the rise of the platform as the dominant infrastructural and economic model of the social web. I have located this moment in the transformation of social network sites into social media platforms. The notion of platformization offers a lens to examine how the infrastructural model of social media platforms is geared towards their extensions into the web, whilst their economic model is based on employing the platform infrastructure to format external web data according to the logic of the platform. I have introduced these two-related processes—the decentralization of platform features and the recentralization of platform ready data—as the double logic of platformization. Tracing this double logic has revealed how social media platforms have introduced platform-native objects such as APIs, social buttons and platform-specific shortened URLs to connect the infrastructural model of the platform to its economic model. I have argued that these platform-native objects serve as prime devices for social media platforms to expand into the web and to create data channels to collect and format external web data. That is, social media platforms are building data-intensive infrastructures to reweave the web for social media. I have put forward that one should focus on the platform-specific features of social media platforms to examine the role of the platform infrastructure in optimizing the web for social media. In this way, analyzing platformization can be seen as a way to critically examine social media optimization.

In tracing the platformization of the web I have not only answered Bogost and Montfort's call for taking platforms as computational infrastructures seriously (2009) but have also addressed Lev Manovich' call for advancing software studies methods (2013) with platform-specific methods for analyzing software platforms. In doing so I have added *platform infrastructure studies* to platform studies and have made a contribution via digital methods (Rogers 2013) to study the spread of social media platforms on the web and its effects.

While Marc Andreessen has shown how to critique individual platforms according to their level of programmability (see chapter 2), the point of this dissertation has been to study and develop a critique of the programmability of social media platforms through their extension into the web. Platform infrastructure studies then is an approach to study a platform's ecosystem.

Further research: Social media and app studies

This dissertation has focused on the impact of social media platforms on the web's infrastructure. Since the start of this dissertation in 2009, social media platforms and the way we engage with internet-based services have changed considerably. One of the most significant changes is the shift from accessing social media platforms through a web-based interface to accessing them through mobile phone applications, commonly referred to as apps. In addition, social media platforms may be accessed through third-party clients or other intermediaries such as scheduling tools for cross-posting social media content (see chapter 3). With the rise of smartphones, many social media platforms have developed mobile websites as well as dedicated apps.[200] In the Spring of 2015 over 84% of Facebook's daily active users did not access Facebook through a web browser on a computer but through an app on a mobile phone or tablet (Facebook Investor Relations 2015). Currently, mobile-only users make up over 40% of all Facebook users (Protalinski 2015).

Social media platforms Facebook and Twitter have further extended their platform infrastructures with new mobile frameworks by creating their own mobile apps and by integrating themselves into the app ecology through other apps. For example, both platforms offer app developers login systems so users can log into their apps with their existing Twitter or Facebook credentials.[201] This can be seen as a move by these platforms to establish themselves as identity providers on the web and within the app space. Both platforms also enable app developers to integrate platform-specific content and functionality such as tweets[202] or a mobile Like button[203] into their apps. Returning to the findings from chapter 5 that Facebook has been building a data-intensive infrastructure on the web through the decentralization of platform features and recentralization of platform data, the introduction of the mobile Like can be seen as a further extension of Facebook into the app ecology. This raises the question whether Facebook is employing similar mechanisms of tracking users within apps as across the web through its Like button (see chapter 5).

[200] A great number of new social media apps have specifically been designed for the mobile phone and only exist as apps. The popular photo-sharing app Instagram for example—which is said to have more active users than Twitter (Lorenzetti 2014)—only introduced a web interface for user profiles and images at user requests after having been a mobile-only app for two years (Instagram 2012). Another social media platform that is very popular with teenagers (Lenhart 2015) that—so far—has only existed as an app is photo messaging app Snapchat. It makes use of the mobile phone camera as well as the user's phone contacts to find and connect to friends to send them so-called 'snaps'.

[201] See: https://dev.twitter.com/twitter-kit/ios/twitter-login and https://developers.facebook.com/products/login [Accessed 2 May 2015].

[202] See: https://dev.twitter.com/twitter-kit/ios/show-tweets [Accessed 2 May 2015].

[203] See: https://developers.facebook.com/blog/post/2014/10/02/mobilelike/ [Accessed 2 May 2015].

Further research should focus on the role of social media platforms in the era of apps in order to analyze what kind of data they are collecting within apps and how they may be reconfiguring the app space. The app ecosystem can be seen an additional space for social media platforms to extend themselves into which raises the question whether this form of platformization operates on the same logic as the platformization of web as analyzed in this dissertation. Taking as a point of departure the methodological insights of this thesis, I suggest that medium-specific inquiries into platforms should be expanded to include apps.

The first line of inquiry seeks to investigate which other strategies Facebook and Twitter employ to use their platform infrastructure to reformat web content to integrate it as valuable data into their databases. A case in point are Twitter Cards, which allow webmasters to attach photos, videos and other media to tweets.[204] They function to "enrich" tweets that contain links with content such as photos, videos, app installs, or a preview of an article.[205] For example, a Twitter Card with a link to a YouTube video shows a video preview in the tweet. In order to use Twitter Cards webmasters have to make their websites platform ready by adding a few lines of code to their pages.[206] Similar to webmasters who are implementing Open Graph tags (see chapter 2) to connect their pages to the Facebook Platform, Twitter also makes use of so-called social meta tags as well as the Open Graph protocol to markup external web content for integration with the Twitter Platform.[207] This is comparable to the practice of search engine optimization where webmasters are optimizing their pages for search engines. What does the practice of social media optimization tell us about the organization of content in the social web? How are apps made platform ready (cf. chapter 2)?

Besides offering webmasters and app developers options to integrate their sites and apps with their platforms, Facebook and Twitter have also developed specific mobile development platforms, so-called software development kits (SDKs), which offer tools for the production, distribution, monitoring and monetization of apps. In April 2013, Facebook acquired Parse and integrated it into Facebook Platform which, according to *Wired Magazine* "gives Facebook a means of tying its social network into all those applications [built with Parse], but it also provides the company with a window into what tens of thousands of people are doing with their mobile phones" (Metz 2013). Similarly, the aptly named Twitter Fabric consists of three software development kits that enable developers to build their apps using the Twitter platform. In doing so, Twitter is not only providing a mobile software development platform but also weaving itself into third-party apps, turning Twitter into a mobile infrastructure. This opens up questions about the circulation of data in mobile apps which increasingly become connected to big social media platforms such as Twitter and Facebook. To answer these questions it is important to devise new app-specific methods to make the invisible

[204] See: https://dev.twitter.com/cards/overview [Accessed 27 April 2015].

[205] Twitter Cards come with Twitter Cards Analytics that contain information about key metrics such as URL clicks, app install attempts and Retweets, see: https://dev.twitter.com/cards/analytics [Accessed 27 April 2015].

[206] See: https://dev.twitter.com/cards/overview [Accessed 27 April 2015].

[207] See: https://dev.twitter.com/cards/getting-started#opengraph [Accessed 27 April 2015].

infrastructures of Twitter and Facebook supporting mobile apps visible. Building on the methods presented in this dissertation to detect and map connections between websites and social media platforms, I wish to develop methods and techniques for analyzing and mapping data flows in apps. What kind of data channels are being opened up between mobile apps and social media platforms? What kind of mobile infrastructures are Facebook and Twitter building?

This draws attention to important architectural aspects of apps which are often described as app silos (cf. chapter 2) since they do not seamlessly link to other apps (Mott 2013). For example, when opening an Instagram link in the Twitter app this link opens in the internal Twitter app browser and not in the installed Instagram app. Twitter and Facebook are both advancing ways to create links between apps so this link does open in the Instagram app. Twitter is pursuing this through the previously mentioned Twitter Cards while Facebook has announced a new link type called App Links to enable "cross-platform, open source, and simple mobile deep-linking" at the 2014 f8 Developers Conference ("App Links" 2015; Lardinois 2015). App Links allow for linking between apps and for linking to specific content within apps. These App Links are also part of Parse and Parse Analytics so mobile developers can track how many users have opened their app from another app (Abernathy 2014). In chapter 3, 4 and 5 I discussed how social media platforms have introduced new link types and practices and have industrialized the hyperlink by turning it into an analytical tool. The introduction of App Links raises questions about the political economy of app interlinking (cf. chapter 4) within the app economy. Whereas hyperlinks for linking to webpages on the web are standardized, deep links for linking to app content in the app ecosystem are not standardized. I addition to Facebook's App Links protocol for deep linking, Twitter[208] and Google[209] have developed their own deep linking mechanisms. If links are the currency of the web (see chapter 3 and 4) and likes can be seen as a platform-native currency of the social web (see chapter 5), can we then consider deep links as the emerging currency of the app ecosystem (Franklin 2014)?

These developments show how social media platforms are integrating themselves into apps (and vice versa), pointing towards a platformization of the app space. Most prominently, apps are supposedly killing the web as well as the mobile web with far more time being increasingly spent using apps than on the mobile web. More specifically as a case in point, with the launch of the Facebook Messenger Platform for building apps that integrate with Facebook's Messenger app, we can observe the reconceptualization of the app as a platform.

These developments show the urgency of continually developing new methods to study the platformization of the web as well as the platformization of the app ecosystem, together

[208] See: https://dev.twitter.com/cards/mobile/deep-linking [Accessed 1 May 2015]

[209] See: https://developers.google.com/app-indexing/webmasters/app [Accessed 1 May 2015]. Google has recently started to use deep links to include app content into its search results (Makino, Jung, and Phan 2015; "App Indexing for Google Search" 2015).

with the consequences. Whilst the rise of apps may introduce a different dynamic of platformization, there are likely shared concerns. In all, the concept of platformization and the methods developed for what I have called platform infrastructure studies provide ways to examine not only a platform's ecosystem on the web and changes to the web's infrastructure. They also provide means to examine the app space, including the platformization of apps as well as an app's ecosystem.

English summary

The web as platform: Data flows in the social web

In October 2004, at the first Web 2.0 conference, Tim O'Reilly rhetorically repositioned the web after the dotcom crash as "the web as platform." With this claim he suggested to understand the web not only as a medium for publishing information but also as a computational development platform for building applications. Not only the web as a whole, but also websites themselves are developed as platforms by offering Application Programming Interfaces, APIs. It is through APIs that websites and most notably social network sites can provide structured access to their data and functionality and be turned into platforms. In this dissertation I trace the transition of social network *sites* into social media *platforms* to examine how social media has altered the web.

The key aim is to develop the concept of "platformization" in order to understand this process from an infrastructural perspective. The platformization of the web refers to the rise of the platform as the dominant infrastructural and economic model of the social web and the consequences of the expansion of social media platforms into other spaces online. Platformization, I argue, rests on the dual logic of social media platforms' expansion into the rest of the web and, simultaneously, their drive to make external web data platform ready. As an infrastructural model, social media platforms provide a technological framework for others to build on which, I argue, is geared towards connecting to and thriving on other websites and their data. Making external web data amenable for their own databases is, so I suggest, central to the economic model of social media platforms. These two processes of decentralizing platform features and recentralizing platform ready data characterize what I call the double logic of platformization. This double logic is operationalized through platform-native objects such as APIs, social buttons and shortened URLs, which connect the infrastructural model of the platform to its economic model. I argue that these platform-native objects serve as prime devices for social media platforms to expand into the web and to create data channels for collecting and formatting external web data to fit the underlying logic of the platform. That is, I show how social media platforms are building data-intensive infrastructures to reweave the web for social media.

This argument is organized around five case studies in which I chronologically trace the platformization of the web and its consequences in terms of 1) the transformation of social network sites into social media platforms 2) the restructuring of the blogosphere and the introduction of new linking practices, 3) the changing nature of the hyperlink from a navigational tool into an analytical tool for data capture, 4) the transformation of the currency of the web from link to like and 5) the boundaries of a website and the end of it as a bounded object. Adopting an approach that combines software studies, platform studies and digital

methods, I analyze the underlying platform infrastructure and platform-native objects of the social web to ask what social media has done to the web. As part of this undertaking, I put forward new methods that I frame as digital methods for platform studies which utilize medium-specific features to explore dynamics of platformization.

In the first chapter on 'The platformization of the web' I provide a detailed material-technical perspective on the development and emergence of what we understand as social media platforms today. I trace how social network sites have become social media platforms by outlining three pre-conditions for platformization: the separation of content and presentation with XML, the modularization of content and features with widgets and interfacing with databases through APIs. Taken together, these aspects turn websites into programmable platforms allowing them to extend beyond their boundaries and establish two-way data flows for data exchanges with third parties. I conceptualize these data channels as 'data pours' that not only transfer data from database to database but also format external web data according to the logic of the platform.

In the second chapter on 'The coming of the platforms', I examine the changing structure of the blogosphere in relation to the rise of social media. This is achieved by reconstructing the historical Dutch blogosphere per year using a collection of archived blogs retrieved from the Internet Archive Wayback Machine between 1999 and 2009. Within this archived collection, I trace how social media platforms introduced new linking practices through widgets and show how these widgets transform the hyperlink structure of the blogosphere. To identify the role of social media in this historical Dutch blogosphere, I develop a method to further examine the types of links between blogs and social media platforms. It becomes apparent that with the rise of social media, bloggers are no longer predominantly linking to establish an interlinked network of blogs, a blogosphere, but increasingly connect to social media and, as a result, weave their blogs into the social media ecosystem.

In the third chapter on 'The algorithmization of the hyperlink', I retain my focus on the changing role of links and examine their advancing commodification by social media platforms. Here I show how platforms use their infrastructure to render the web-native object of the hyperlink into a platform-specific shortened URL. In doing so, social media platforms change the function of the link from a navigational into an analytical device amenable for data capture. Social buttons play a central role in this, as they create new forms of automated, data-rich shortened URLs that are formatted to fit the purpose of the platform; that is, to feed the underlying algorithms and analytics suites. In the web as platform, the link becomes a database call and a device to make external web data 'platform ready.' I develop a method to examine the actors involved in this reconfiguration of the hyperlink by following shortened URLs.

The fourth chapter on 'The Like economy' shows how social media platforms employ social buttons as part of their technical infrastructure to turn social activities into valuable data, conceptualized as a so-called 'Like economy'. I contextualize the rise of social buttons as metrics for user engagement and link them to different web economies: the hit, link and Like economy. I explore how the platformization of the web shifts the currency of the web from web-native links to platform-native likes which are tied not to the web at large, but to the mechanics and logics of specific social media platforms. Facebook's Like economy is enabled by the interconnected dynamics of the decentralization of data production—by offering social buttons to like content across the web—and the recentralization of data collection through these buttons. I devise a method to map the presence of social buttons on a collection of websites to show how they create new forms of connectivity between websites beyond hyperlinks, introducing an alternative fabric of the web.

In the final chapter on 'Website ecologies', I explore the changing boundaries of the website in the web as platform. Websites are increasingly shaped by and assembled from content and functionality such as embedded content, social plugins and advertisements, thereby complicating the notion of the website as a bounded object. The third-party objects present on websites draw attention to the larger techno-commercial configurations of the web that these sites are embedded in. I therefore suggest to reconceptualize the study of websites as website ecology which analyzes how various relations between the different actors on the web have become inscribed in a website's source code. In this chapter, I propose a method that uses the source code of an archived website to study a website's ecosystem over time as a way to examine the spread of platformization. In addition, I employ the affordances of social media platform APIs to retrieve missing platform content in archived websites.

The five case studies demonstrate that the consequences of social media platforms' tight integration with the web—platformization—typify a significant change in how the web's infrastructure is put to use. To study the platformization of the web, I therefore argue, one should engage with data exchange mechanisms, new means to connect websites, the transformation and commodification of the hyperlink, the introduction of new web currencies for web content such likes, shares and retweets, and the redrawn boundaries of the website. That is, one should recognize the platform-specific objects that have been introduced by social media platforms that take on various social and technical functions, one of them being to reweave the fabric of the web.

In this dissertation, I develop a platform critique that revolves around the notion of platformization that is positioned as a contribution to the emerging fields of software studies and platform studies, and draws on digital methods to study the effects of social media on the web's infrastructure. In doing so, I answer current calls for taking platforms as computational infrastructures seriously, and respond to the need for new methodological development to advance the fields of software studies and platform studies. Ultimately, I propose a new branch of platform studies that I call platform infrastructure studies, which analyzes the ecosystem of software platforms with platform-specific digital methods.

In the conclusion, moreover, I ask whether this sort of critique still applies with social media access shifting from the web to mobile apps. In presenting a future research agenda, I address a number of developments that show how social media platforms are integrating themselves into apps (and vice versa), pointing towards a platformization of the app space.

Most prominently, apps are supposedly killing the web as well as the mobile web with far more time being increasingly spent using apps than on the mobile web. More specifically as a case in point, with the launch of the Facebook Messenger Platform for building apps that integrate with Facebook's Messenger app, we can observe the reconceptualization of the app as a platform.

These developments show the urgency of continually developing new methods to study the platformization of the web as well as the platformization of the app ecosystem, together with the consequences. Whilst the rise of apps may introduce a different dynamic of platformization, there are likely shared concerns. In all, the concept of platformization and the methods developed for what I have called platform infrastructure studies provide ways to examine not only a platform's ecosystem on the web but also changes to the web's infrastructure. They also provide means to examine the app space, including the platformization of apps as well as an app's ecosystem.

Nederlandse samenvatting

Het web als platform: Datastromen in het sociale web

In oktober 2004, tijdens de eerste Web 2.0 conferentie, herpositioneerde Tim O'Reilly het web na de dot-com crash retorisch als "het web als platform". Hiermee stelde hij voor om het web niet alleen te begrijpen als een medium voor het publiceren van informatie, maar ook als een computerplatform om applicaties voor te ontwikkelen. Naast het web als geheel, kunnen ook websites *zelf* ontwikkeld worden als platformen door het aanbieden van *Application Programming Interfaces*, API's. Door middel van API's kunnen websites, en in het bijzonder sociale netwerksites, gestructureerde toegang tot hun data en functionaliteit aanbieden en daarmee platformen worden. In dit proefschrift traceer ik de transitie van sociale netwerk*sites* naar sociale media*platformen* om te bestuderen hoe sociale media het web hebben veranderd.

Ik ontwikkel het begrip 'platformisatie' om dit proces vanuit een infrastructureel perspectief te begrijpen. De platformisatie van het web verwijst naar de opkomst van het platform als het dominante infrastructurele en economische model van het sociale web en naar de gevolgen van de verspreiding van sociale mediaplatformen online. Ik betoog dat platformisatie berust op de dubbele logica van de verspreiding van sociale mediaplatformen door de rest van het web en hun ambitie om externe webdata 'platformklaar' te maken. Als een infrastructureel model bieden platformen een technologische basis voor andere partijen om zich verder op te ontwikkelen. Dit model richt zich op het aansluiten en profiteren van andere websites en hun data. Ik stel dat het gereedmaken van externe webdata voor hun eigen databases centraal staat in het economische model van sociale mediaplatformen. Deze twee processen van het decentraliseren van platformfuncties en het recentraliseren van platformklare data karakteriseren wat ik de dubbele logica van platformisatie noem. Deze dubbele logica wordt geoperationaliseerd door middel van platform-specifieke objecten zoals API's, *social buttons* en verkorte URLs, die het infrastructurele model van het platform met het economische model verbinden. Deze platform-specifieke objecten zijn de belangrijkste elementen voor sociale mediaplatformen om zich door het web te verspreiden en om datakanalen te creëren voor het verzamelen en formatteren van externe webdata en vervolgens deze data aan te passen aan de onderliggende logica van het platform. Hiermee laat ik zien hoe sociale mediaplatformen data-intensieve infrastructuren bouwen, die volledig gericht zijn op sociale media, om het web opnieuw in elkaar 'weven'. Dit argument wordt uitgewerkt in vijf case studies waarin ik in chronologische volgorde de platformisatie van het web en de consequenties daarvan beschrijf voor 1) de transformatie van sociale netwerksites naar sociale mediaplatformen, 2) de herstructurering van de blogosfeer en de introductie van nieuwe linkpraktijken, 3) de veranderende aard van de hyperlink van een navigatie-instrument naar een analytisch instrument om data te verzamelen, 4) de transformatie van de valuta van het web van links naar likes, en 5) de grenzen van de website en het einde van de website als een begrensd object. Door middel van een gecombineerde software studies-, platform studies- en digital methods benadering analyseer ik de onderliggende platforminfrastructuur en de platform-specifieke elementen van het sociale web om in kaart te brengen wat sociale media met het web gedaan hebben. Hiervoor heb ik nieuwe analysemethoden ontwikkeld die ik zie als digitale methoden voor platformstudies waarbij ik gebruik maak van de medium-specifieke eigenschappen van sociale media om de dynamiek van platformisatie te verkennen.

In het eerste hoofdstuk over 'De platformisatie van het web' bied ik een gedetailleerd materiaal-technisch perspectief op de ontwikkeling en opkomst van wat wij tegenwoordig zien als sociale mediaplatformen. Ik traceer hoe sociale netwerk*sites* sociale media*platformen* zijn geworden door een overzicht te bieden van de drie voorwaarden voor platformisatie: de scheiding van inhoud en presentatie aan de hand van XML, de modularisering van inhoud en functies aan de hand van widgets en het communiceren met databases door middel van API's. Deze elementen veranderen websites in programmeerbare platformen waardoor ze zichzelf door het web kunnen verspreiden en kanalen kunnen opzetten voor de uitwisseling van gegevens met derde partijen. Ik conceptualiseer deze zogenaamde gegevenskanalen als *data pours* die niet slechts data van database naar database transporteren, maar ook externa data aanpassen en formatteren volgens de logica van het platform.

In het tweede hoofdstuk over 'De komst van de platformen' onderzoek ik de veranderende structuur van de blogosfeer met betrekking tot de opkomst van sociale media. Dit doe ik door middel van een reconstructie van de historische Nederlandse blogosfeer per jaar tussen 1999 en 2009 met behulp van een verzameling van gearchiveerde blogs die ik heb verkregen uit het *Internet Archive Wayback Machine*. Ik gebruik deze verzameling om te onderzoeken hoe sociale mediaplatformen nieuwe linkpraktijken hebben geïntroduceerd door middel van widgets en hoe deze widgets de hyperlinkstructuur van de blogosfeer hebben veranderd. Ik ontwikkel een methode om de relatie tussen blogs en sociale mediaplatformen verder te onderzoeken en op deze manier na te gaan welke rol sociale media in de geschiedenis van de Nederlandse blogosfeer hebben gespeeld. Deze analyse toont aan dat met de opkomst van sociale media, bloggers niet langer hoofdzakelijk naar andere blogs linken en hiermee een onderling verbonden netwerk van blogs—een blogosfeer—creëren, maar dat ze in toenemende mate naar sociale media linken en als gevolg hiervan zichzelf in het sociale media-ecosysteem weven.

In het derde hoofdstuk over 'De algoritmisatie van de hyperlink' kijk ik verder naar de veranderende rol van links en onderzoek ik de vercommercialisering van links door sociale mediaplatformen. Ik laat zien hoe platformen gebruik maken van hun infrastructuur om de web-eigen hyperlink te veranderen in een platform-eigen verkorte URL. Hiermee veranderen sociale mediaplatformen de functie van de link van een instrument voor navigatie in een analytisch apparaat waarmee data verzameld kan worden. Social buttons spelen hierbij een belangrijke rol, omdat ze nieuwe vormen van geautomatiseerde, informatierijke, verkorte URLs creëren die speciaal geformatteerd zijn voor het doel van het platform. Hiermee voeden deze nieuwe verkorte URLs en de data die ze verzamelen de onderliggende algoritmes en analysesystemen van het platform. In het web als platform is de link een zogenaamde *database call* geworden en daarmee een instrument om externe webdata 'platform-klaar' te maken. In dit hoofdstuk ontwikkel ik een methode om te onderzoeken wie er betrokken zijn bij deze vercommercialisering van de hyperlink door het pad van verkorte URL's te traceren.

Het vierde hoofdstuk over 'De Like-economie' laat zien hoe sociale mediaplatformen hun social buttons als onderdeel van hun technische infrastructuur gebruiken om sociale activiteiten om te zetten in waardevolle gegevens, wat ik aanmerk als de zogeheten 'Like-economie'. Ik beschrijf de opkomst van social buttons in relatie tot eerdere maatstaven om de betrokkenheid van gebruikers te meten en koppel deze aan verschillende webeconomieën: de hit, link en Like-economie. Ik onderzoek hoe de platformisatie van het web de valuta van web-eigen links naar platform-eigen likes, shares en tweets heeft verschoven. Deze valuta zijn niet langer verbonden aan het web in het algemeen maar aan de infrastructuur en logica van specifieke sociale mediaplatformen. Ik beschrijf hoe Facebooks Like-economie mogelijk wordt gemaakt door de dynamiek van decentralisatie en recentralisatie. Aan de ene kant decentraliseren social buttons de productie van data en aan de andere kant recentraliseren ze de verzameling van data. Ik ontwerp een methode om de aanwezigheid van social buttons in een verzameling van websites in kaart te brengen. Hiermee toon ik aan hoe social buttons, ten opzichte van traditionele hyperlinks, nieuwe verbindingen tussen websites creëren en daarmee een alternatieve structuur van het web weven.

In het laatste hoofdstuk over 'Website-ecologie' verken ik de veranderende grenzen van de website in het web als platform. Websites worden steeds meer gevormd door en samengesteld uit de inhoud en functionaliteit van derde partijen. Ze bevatten bijvoorbeeld ingesloten foto's en video's van andere websites, sociale plugins en advertenties waardoor de notie van de website als een begrensd object gecompliceerd wordt. Hiermee wordt de aandacht gevestigd op de techno-commerciële configuraties van het web waarin deze websites zich bevinden. Ik stel dan ook voor de studie van websites te benaderen als website-ecologie, waarin gekeken wordt naar de verschillende relaties tussen de actoren op het web. Deze relaties zijn terug te vinden in de *source code* van de websites. In dit hoofdstuk ontwikkel ik een methode die gebruik maakt van de broncode van een gearchiveerde website om het veranderende ecosysteem van een website in een bepaalde periode in kaart te brengen. Dit presenteer ik als een manier om de verspreiding van platformisatie te onderzoeken vanuit het perspectief van de website. Daarnaast heb ik in dit hoofdstuk gebruik gemaakt van de mogelijkheden van de API's van sociale mediaplatformen om de inhoud van deze platformen, zoals deze getoond wordt op externe websites en niet gearchiveerd is, op te sporen en opnieuw te reconstrueren.

De vijf case studies tonen aan dat nauwe integratie van sociale mediaplatformen met het web, oftewel platformisatie, een belangrijke verandering teweeg brengt in de manier waarop de infrastructuur van het web gebruikt wordt. Om de platformisatie van het web te bestuderen, pleit ik daarom voor het bestuderen van 1) data-uitwisselingsmechanismen, 2) nieuwe vormen van verbindingen tussen websites, 3) de transformatie en vercommercialisering van de hyperlink, 4) de introductie van nieuwe web valuta voor web content zoals de like, de share en de retweet, en 5) de veranderende grenzen van de website. Het is hierbij belangrijk om de platform-specifieke objecten die door sociale mediaplatformen zijn geïntroduceerd te herkennen. De objecten hebben namelijk verschillende sociale en technische functies waaronder het opnieuw structureren van het web.

In dit proefschrift ontwikkel ik een platformkritiek rondom het begrip ‘platformisatie’ om de effecten van sociale media op de infrastructuur van het web te bestuderen. Ik positioneer dit als een bijdrage, gebaseerd op digital methods, aan de opkomende vakgebieden van software studies en platform studies. Daarmee beantwoord ik oproepen uit het veld om de computationele infrastructuur van platformen serieus te nemen en om methodologische ontwikkeling op het gebied van software studies en platform studies verder te bevorderen. Uiteindelijk stel ik een nieuwe tak van platform studies voor die ik *platform infrastructure studies* noem: een benadering die het ecosysteem van software platformen bestudeert met platform-specifieke digitale methoden.

In de conclusie stel ik de vraag of dit soort platformkritiek nog steeds geldt met de verschuiving van sociale media van het web naar mobiele apps. Ik presenteer een toekomstige onderzoeksagenda waarin ik een aantal ontwikkelingen bespreek die laten zien hoe sociale mediaplatformen steeds verder integreren met apps (en vice versa), wat wijst op de platformisatie van het app-ecosysteem. In het bijzonder, zo stelt men, dragen apps bij aan het einde van het web en met name het mobiele web aangezien 86% van de tijd op mobiele apparaten in apps wordt doorgebracht en slechts 14% van deze tijd op het mobiele web (Khalaf 2014). We kunnen tevens zien hoe apps ook zelf platformen worden, met bijvoorbeeld de lancering van het Facebook Messenger Platform waarmee ontwikkelaars apps kunnen bouwen die integreren met de Facebook Messenger app (Franklin 2015).

Deze ontwikkelingen tonen aan hoe belangrijk het is om voortdurend methoden te ontwikkelen om zowel de platformisatie van het web als de platformisatie van het app-ecosysteem te bestuderen, evenals de consequenties ervan. Terwijl de opkomst van apps mogelijk een andere dynamiek van platformisatie introduceert, heb ik tevens aangetoond dat er gedeelde aanknopingspunten zijn voor de bestudering ervan. Samenvattend bieden het concept van platformisatie en de onderzoeksmethoden die ik heb ontwikkeld voor wat ik *platform infrastructure studies* noem, manieren om niet alleen het ecosysteem van een platform op het web te bestuderen maar ook de veranderingen in de infrastructuur van het web zelf. Daarnaast is het een vruchtbare benadering om de app omgeving te onderzoeken, inclusief de platformisatie van apps alsook hun ecosystemen.

References

Abernathy, Christine. 2014. “Announcing Parse Analytics Integration with App Links.” *Parse Blog*. August 21. http://blog.parse.com/announcements/announcing-parse-analytics-integration-with-app-links/.

Acar, Güneş, Christian Eubank, Steven Englehardt, Marc Juarez, Arvind Narayanan, and Claudia Diaz. 2014. “The Web Never Forgets: Persistent Tracking Mechanisms in the Wild.” https://securehomes.esat.kuleuven.be/~gacar/persistent/.

Acar, Güneş, Brendan Van Alsenoy, Frank Piessens, Claudia Diaz, and Bart Preneel. 2015. “Facebook Tracking Through Social Plug-Ins.”

Adamic, Lada A., and Natalie Glance. 2005. “The Political Blogosphere and the 2004 US Election: Divided They Blog.” In *Proceedings of the 3rd International Workshop on Link Discovery*, 36–43. ACM. http://dl.acm.org/citation.cfm?id=1134277.

Allen, Matthew. 2013. “What Was Web 2.0? Versions as the Dominant Mode of Internet History.” *New Media & Society* 15 (2): 260–75. doi:10.1177/1461444812451567.

Amazon. 2002. “Amazon.com Launches Web Services.” Press Release. Seattle. http://phx.corporate-ir.net/phoenix.zhtml?c=176060&p=irol-newsArticle&ID=503034.

Ammann, Rudolf. 2009. “Blogosphere 1998: Analysis.” *Tawawa*. November 5. http://tawawa.org/ark/2009/11/5/blogosphere-1998-analysis.html.

———. 2011. “Reciprocity, Social Curation and the Emergence of Blogging: A Study in Community Formation.” *Procedia - Social and Behavioral Sciences* 22: 26–36. doi:10.1016/j.sbspro.2011.07.053.

Anderson, Chris. 2006. *The Long Tail: Why the Future of Business Is Selling Less of More*. Hyperion.

———. 2009. *Free: The Future of a Radical Price*. New York, NY: Hyperion.

Anderson, Chris, and Michael Wolff. 2010. “The Web Is Dead. Long Live the Internet.” *WIRED*. August 17. http://www.wired.com/2010/08/ff_webrip/all/1.

Anderson, Paul. 2012. *Web 2.0 and Beyond: Principles and Technologies*. Boca Raton, FL: CRC Press.

Andreessen, Marc. 2007a. “Analyzing the Facebook Platform, Three Weeks in.” *Blog.pmarca.com*. June 12. https://web.archive.org/web/20071002070223/http://blog.pmarca.com/2007/06/analyzing_the_f.html.

———. 2007b. “The Three Kinds of Platforms You Meet on the Internet.” *Blog.pmarca.com*. September 16. https://web.archive.org/web/20071002031605/http://blog.pmarca.com/2007/09/the-three-kinds.html.

Ankerson, Megan Sapnar. 2009. “Historicizing Web Design: Software. Style, and the Look of the Web.” In *Convergence, Media, History*, edited by Janet Staiger and Sabine Hake, 192. New York, NY: Routledge.

———. 2012. “Writing Web Histories with an Eye on the Analog Past.” *New Media & Society* 14 (3): 384–400.

Antoniades, Demetris, Iasonas Polakis, Georgios Kontaxis, Elias Athanasopoulos, Sotiris Ioannidis, Evangelos P. Markatos, and Thomas Karagiannis. 2011. “We.b: The Web of Short URLs.” In *Proceedings of the 20th International Conference on World Wide Web*, 715–24. http://dl.acm.org/citation.cfm?id=1963505.

AOL Annual Report. 1995. “America Online 1995 Annual Report.” http://archive.org/details/09Kahle002309.

Appelquist, Daniel, Dan Brickley, Melvin Carvahlo, and Renato Iannella. 2010. “A Standards-Based, Open and Privacy-Aware Social Web.” W3C Incubator Group Report 6th December 2010. W3C Incubator Group Report. http://www.w3.org/2005/Incubator/socialweb/XGR-socialweb-20101206/.

Apperley, Thomas, and Darshana Jayemane. 2012. “Game Studies’ Material Turn.” *Westminster Papers in Communication and Culture* 9 (1): 5–26.

“App Indexing for Google Search.” 2015. *Google Developers*. April 16. https://developers.google.com/app-indexing/.

“App Links.” 2015. Accessed May 4. http://applinks.org/.

Arora, Payal. 2014. “Walled Gardens.” In *The Leisure Commons: A Spatial History of Web 2.0*, 378. Hoboken: Taylor and Francis.

Arrington, Michael. 2008. “Facebook Isn’t A Social Network. And Stop Trying to Make New Friends There.” *TechCrunch*. September 15. http://techcrunch.com/2008/09/15/facebook-isnt-a-social-network-and-dont-try-to-make-new-friends-there/.

Arthur, W. Brian. 1989. “Competing Technologies, Increasing Returns, and Lock-In by Historical Events.” *The Economic Journal* 99 (394): 116–31. doi:10.2307/2234208.

Arvidsson, Adam. 2011. “General Sentiment: How Value and Affect Converge in the Information Economy.” *The Sociological Review* 59 (s2): 39–59.

Aufderheide, Patricia. 2002. “Competition and Commons: The Public Interest in and after the AOL-Time Warner Merger.” *Journal of Broadcasting & Electronic Media* 46: 518.

Bakshy, Eytan. 2012. “Rethinking Information Diversity in Networks.” January 17. https://www.facebook.com/notes/facebook-data-team/rethinking-information-diversity-in-net%ADworks/10150503499618859.

Bakshy, Eytan, Solomon Messing, and Lada Adamic. 2015. “Exposure to Ideologically Diverse News and Opinion on Facebook.” *Science*, May. doi:10.1126/science.aaa1160.

Baldwin, Carliss Young, and Kim B. Clark. 2000. *Design Rules: The Power of Modularity*. MIT Press.

Bastian, Mathieu, Sebastien Heymann, Mathieu Jacomy, and others. 2009. “Gephi: An Open Source Software for Exploring and Manipulating Networks.” *ICWSM* 8: 361–62.

Batini, C., E. Nardelli, and R. Tamassia. 1986. “A Layout Algorithm for Data Flow Diagrams.” *IEEE Transactions on Software Engineering* SE-12 (4): 538–46. doi:10.1109/TSE.1986.6312901.

Baym, Nancy. 2007. “The Widgetized Self.” *Online Fandom*. April 9. http://www.onlinefandom.com/archives/the-widgetized-self/.

Bechmann, Anja. 2013. "Internet Profiling: The Economy of Data Intraoperability on Facebook and Google." *MedieKultur. Journal of Media and Communication Research* 29 (55). http://www.nys.dk/index.php/mediekultur/article/view/8070.

Beer, David. 2009. "Power through the Algorithm? Participatory Web Cultures and the Technological Unconscious." *New Media & Society* 11 (6): 985–1002. doi:10.1177/1461444809336551.

———. 2013. *Popular Culture and New Media: The Politics of Circulation*. New York: Palgrave Macmillan.

Beer, David, and Roger Burrows. 2007. "Sociology And, of and in Web 2.0: Some Initial Considerations." *Sociological Research Online* 12 (5). doi:10.5153/sro.1560.

Ben-David, Anat. 2011. "The Emergence of the Palestinian Web-Space: A Digital History of a Digital Landscape." In *MiT7 Unstable Platforms: The Promise and Peril of Transition*. Cambridge, MA. http://web.mit.edu/comm-forum/mit7/papers/MIT7_bendavid.pdf.

Ben-David, Anat, and Hugo Huurdeman. 2014. "Web Archive Search as Research: Methodological and Theoretical Implications." *Alexandria* 25 (1): 93–111.

Benkler, Yochai. 2006. *The Wealth of Networks: How Social Production Transforms Markets and Freedom*. New Haven: Yale University Press.

Benkler, Yochai, and Aaron Shaw. 2010. "A Tale of Two Blogospheres: Discursive Practices on the Left and Right." *Berkman Center for Internet and Society Working Paper Series*. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.170.2648&rep=rep1&type=pdf.

Benslimane, Djamal, Schahram Dustdar, and Amit Sheth. 2008. "Services Mashups: The New Generation of Web Applications." *IEEE Internet Computing*.

Berners-Lee, Tim. 1996. "Declaration Presented by Tim Berners-Lee." February 28. http://www.w3.org/People/Berners-Lee/9602affi.html.

———. 1997. "Links and Law." *Commentary on Web Architecture*. April. http://www.w3.org/DesignIssues/LinkLaw.html.

———. 1998. "Cool URIs Don't Change." *W3C*. http://www.w3.org/Provider/Style/URI.html.

———. 2000. *Weaving the Web: The Original Design and Ultimate Destiny of the World Wide Web*. 1st ed. New York, NY, USA: HarperBusiness.

———. 2010. "Long Live the Web: A Call for Continued Open Standards and Neutrality." *Scientific American*, December. http://www.scientificamerican.com/article/long-live-the-web/.

Berners-Lee, Tim, Robert Cailliau, Ari Luotonen, Henrik Frystyk Nielsen, and Arthur Secret. 1994. "The World-Wide Web." *Communications of the ACM* 37 (8): 76–82. doi:10.1145/179606.179671.

Berry, David M. 2011. *The Philosophy of Software: Code and Mediation in the Digital Age*. Basingstoke, UK: Palgrave Macmillan.

———. 2012. "Life in Code and Software." *Living Books About Life*. September 23. http://www.livingbooksaboutlife.org/books/Life_in_Code_and_Software/Introduction.

Birrell, Andrew D., and Bruce Jay Nelson. 1984. "Implementing Remote Procedure Calls." *ACM Transactions on Computer Systems* 2 (1): 39–59. doi:10.1145/2080.357392.

Blood, Rebecca. 2004. “How Blogging Software Reshapes the Online Community.” *Communications of the ACM* 47 (12): 53–55.

Bodle, Robert. 2011. “Regimes of Sharing.” *Information, Communication & Society* 14 (3): 320–37. doi:10.1080/1369118X.2010.542825.

Bogost, Ian, and Nick Montfort. 2007. “New Media as Material Constraint: An Introduction to Platform Studies.” In *Electronic Techtonics: Thinking at the Interface*, 1–14. Duke University.

———. 2009. “Platform Studies: Frequently Questioned Answers.” In *Digital Arts and Culture*. http://escholarship.org/uc/item/01r0k9br.pdf.

Bojars, Uldis, John Breslin, and Stefan Decker. 2008. “Social Networks and Data Portability Using Semantic Web Technologies.” http://vmserver14.nuigalway.ie/xmlui/handle/10379/398.

boyd, danah. 2006. “A Blogger’s Blog: Exploring the Definition of a Medium.” *Reconstruction* 6 (4). http://www.danah.org/papers/ABloggersBlog.pdf.

———. 2007a. “About Those Walled Gardens.” *Apophenia*. February 5. http://www.zephoria.org/thoughts/archives/2007/02/05/about_those_wal.html.

———. 2007b. “Viewing American Class Divisions through Facebook and MySpace.” *Danah Boyd*. June 24. http://www.danah.org/papers/essays/ClassDivisions.html.

———. 2010. “Social Network Sites as Networked Publics: Affordances, Dynamics, and Implications.” In *A Networked Self: Identity, Community, and Culture on Social Network Sites*, edited by Zizi Papacharissi, 39–58. New York: Routledge.

boyd, danah, and Nicole Ellison. 2008. “Social Network Sites: Definition, History, and Scholarship.” *Journal of Computer-Mediated Communication* 13 (1): 210–30.

Braiker, Brian. 2006. “Tech: Welcome, Year of the Widget.” *Newsweek*. December 21. http://www.newsweek.com/tech-welcome-year-widget-105589.

Brin, Sergey, and Larry Page. 1998. “The Anatomy of a Large-Scale Hypertextual Web Search Engine.” *Computer Networks and ISDN Systems* 30 (1): 107–17.

Bross, Justus, Matthias Quasthoff, Philipp Berger, Patrick Hennig, and Christoph Meinel. 2010. “Mapping the Blogosphere with Rss-Feeds.” In *Advanced Information Networking and Applications (AINA), 2010 24th IEEE International Conference on*, 453–60. IEEE. http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5474737.

Brügger, Niels. 2008. “The Archived Website and Website Philology: A New Type of Historical Document?” *Nordicom Review* 29 (2).

———. 2009. “Website History and the Website as an Object of Study.” *New Media & Society* 11 (1-2): 115–32.

———. 2013. “Web Historiography and Internet Studies: Challenges and Perspectives.” *New Media & Society* 15 (5): 752–64.

Brunelle, Justin F., Mat Kelly, Michele C. Weigle, and Michael L. Nelson. 2015. “The Impact of JavaScript on Archivability.” *International Journal on Digital Libraries*, January, 1–23. doi:10.1007/s00799-015-0140-8.

Bruns, Axel. 2007. “Methodologies for Mapping the Political Blogosphere: An Exploration Using the IssueCrawler Research Tool.” *First Monday* 12 (5). http://firstmonday.org/ojs/index.php/fm/article/viewArticle/1834.

———. 2008a. *Blogs, Wikipedia, Second Life, and Beyond: From Production to Produsage*. New York: Peter Lang.

———. 2008b. “Who Controls the Means of Produsage?” *Re-Public: Re-Imagining Democracy*. http://eprints.qut.edu.au/12302/.

Bruns, Axel, Lars Kirchhoff, and Thomas Nicolai. 2009. “Mapping the Australian Political Blogosphere.” http://journal.webscience.org/234/.

Bruza, Peter, and Theo Van der Weide. 1993. “The Semantics of Data Flow Diagrams.” In *Proceedings of the International Conference on Management of Data*, 66–78. Hyderabad, India: McGraw-Hill Publishing Company.

Bucher, Taina. 2012a. “A Technicity of Attention: How Software ‘Makes Sense.’” *Culture Machine* 13: 1–13.

———. 2012b. “Programmed Sociality: A Software Studies Perspective on Social Networking Sites.” Oslo, Norway: University of Oslo. http://tainabucher.com/wp-content/uploads/2009/08/Bucher_Ph.D.diss_.pdf.

———. 2012c. “Want to Be on the Top? Algorithmic Power and the Threat of Invisibility on Facebook.” *New Media & Society* 14 (7): 1164–80. doi:10.1177/1461444812440159.

———. 2013. “Objects of Intense Feeling: The Case of the Twitter API.” *Computational Culture* 3 (November). http://computationalculture.net/article/objects-of-intense-feeling-the-case-of-the-twitter-api.

BuildWith. 2014. “Facebook Like Usage Statistics.” August 28. http://trends.builtwith.com/widgets/Facebook-Like.

———. 2015. “Facebook for Websites Usage Statistics.” *Websites Using Facebook for Websites*. March 4. http://trends.builtwith.com/javascript/Facebook-for-Websites.

Chaabane, Abdelberi, Yuan Ding, Ratan Dey, Mohamed Ali Kaafar, and Keith W. Ross. 2014. “A Closer Look at Third-Party OSN Applications: Are They Leaking Your Personal Information?” In *Passive and Active Measurement*, edited by Michalis Faloutsos and Aleksandar Kuzmanovic, 235–46. Lecture Notes in Computer Science 8362. Springer International Publishing. http://link.springer.com/chapter/10.1007/978-3-319-04918-2_23.

Cheng, Xu, Cameron Dale, and Jiangchuan Liu. 2008. “Statistics and Social Network of YouTube Videos.” In *Quality of Service, 2008. IWQoS 2008. 16th International Workshop on*, 229–38. IEEE. http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4539688.

Chhabra, Sidharth, Anupama Aggarwal, Fabricio Benevenuto, and Pponnurangam Kumaraguru. 2011. “Phi. Sh/$ oCiaL: The Phishing Landscape through Short URLs.” In *Proceedings of the 8th Annual Collaboration, Electronic Messaging, Anti-Abuse and Spam Conference*, 92–101. http://dl.acm.org/citation.cfm?id=2030387.

Chia, Aleena. 2012. “Welcome to Me-Mart The Politics of User-Generated Content in Personal Blogs.” *American Behavioral Scientist* 56 (4): 421–38.

Chi, Ed H., James Pitkow, Jock Mackinlay, Peter Pirolli, Rich Gossweiler, and Stuart K. Card. 1998. “Visualizing the Evolution of Web Ecologies.” In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 400–407. ACM Press/Addison-Wesley Publishing Co. http://dl.acm.org/citation.cfm?id=274699.

Chun, Wendy Hui Kyong. 2011. *Programmed Visions: Software and Memory*. Cambridge, MA: The MIT Press.

Clark, Jessica, Nick Couldry, Abigail T. De Kosnik, Tarleton Gillespie, Henry Jenkins, Christopher Kelty, Zizi Papacharissi, Alison Powell, and José van Dijck. 2014.

"Participations| Part 5: PLATFORMS." *International Journal of Communication* 8 (0): 28.

comScore. 2011. "The Netherlands Ranks #1 Worldwide in Penetration for Twitter and LinkedIn." London, UK. http://www.comscore.com/Insights/Press-Releases/2011/4/The-Netherlands-Ranks-number-one-Worldwide-in-Penetration-for-Twitter-and-LinkedIn.

Connell, Derrick. 2013. "Evolving Search on Facebook." *Bing Blogs. Search Blog*. January 15. http://blogs.bing.com/search/2013/01/15/evolving-search-on-facebook/.

Coté, Mark, and Jennifer Pybus. 2007. "Learning to Immaterial Labour 2.0: MySpace and Social Networks." *Ephemera: Theory & Politics in Organization* 7 (1): 88–106.

Cubrilovic, Nik. 2011. "Logging out of Facebook Is Not Enough." *New Web Order*. September 25. https://www.nikcub.com/posts/logging-out-of-facebook-is-not-enough-2/.

Cutts, Matt, and Jason Shellen. 2005. "Preventing Comment Spam." *Official Google Blog*. January 19. http://googleblog.blogspot.nl/2005/01/preventing-comment-spam.html.

D'Alessio, Dave. 1997. "Use of the World Wide Web in the 1996 US Election." *Electoral Studies* 16 (4): 489–500. doi:10.1016/S0261-3794(97)00044-9.

D'Angelo, Frank. 2009. "Happy Birthday, Digital Advertising!" *DigitalNext: A Blog on Emerging Media and Technology. Ad Age.* October 26. http://adage.com/article/digitalnext/happy-birthday-digital-advertising/139964/.

Dash, Anil. 2012. "The Web We Lost." *Anil Dash. A Blog about Making Culture*. December 13. http://dashes.com/anil/2012/12/the-web-we-lost.html.

David, Paul. 1985. "Clio and the Economics of QWERTY." *The American Economic Review* 75 (2): 332–37.

Dean, Jodi. 2005. "Communicative Capitalism: Circulation and the Foreclosure of Politics." *Cultural Politics* 1 (1): 51–74.

De Cristofaro, Emiliano, Arik Friedman, Guillaume Jourjon, Mohamed Ali Kaafar, and M. Zubair Shafiq. 2014. "Paying for Likes? Understanding Facebook Like Fraud Using Honeypots." *arXiv:1409.2097 [physics]*, September. http://arxiv.org/abs/1409.2097.

@delbius. 2010. "Trust And Safety." *Twitter Blog*. March 10. https://blog.twitter.com/2010/trust-and-safety.

De Maeyer, Juliette. 2011. "How to Make Sense of Hyperlinks? An Overview of Link Studies." In *A Decade in Internet Time: Symposium on the Dynamics of Internet and Society*. http://papers.ssrn.com/sol3/papers.cfm?abstract_id=1926301.

———. 2013. "Towards a Hyperlinked Society: A Critical Review of Link Studies." *New Media & Society* 15 (5): 737–51. doi:10.1177/1461444812462851.

DeMarco, Tom. 1978. *Structured Analysis and System Specification*. 1 edition. New York, NY: Yourdon.

De Souza, Cleidson, Jon Froehlich, and Paul Dourish. 2005. "Seeking the Source: Software Source Code as a Social and Technical Artifact." In *Proceedings of the 2005 International ACM SIGGROUP Conference on Supporting Group Work*, 197–206. ACM. http://dl.acm.org/citation.cfm?id=1099239.

Deswaef, Julien. 2014. "[loveMachine] Automatic like Processor." *Xuv's Notebook*. May 4. http://w.xuv.be/projects/love_machine.

Digg. 2006. “Digg External Story Submission Process.” http://videos.revision3.com/diggnation/digg_submission_062506.pdf.

Dunning, John. 2015. “Yahoo Mail Contact Cards Updated with LinkedIn & Twitter.” *Yahoo! Blog*. April 17. http://yahoo.tumblr.com/post/116559201039/yahoo-mail-contact-cards-updated-with-linkedin.

Duray, Dan. 2011. “The End of Blogging.” *New York Observer*. February 1. http://observer.com/2011/02/the-end-of-blogging/.

DuVander, Adam. 2012. “Over 2,000 APIs Added in 2011: Social, Telephony, Open Government.” *ProgrammableWeb*. January 4. http://www.programmableweb.com/news/over-2000-apis-added-2011-social-telephony-open-government/2012/01/04.

Elmer, Greg. 2002. “The Case of Web Browser Cookies: Enabling/disabling Convenience and Relevance on the Web.” In *Critical Perspectives on the Internet*. Lanham, MD: Rowman & Littlefield.

———. 2006. “Re-Tooling the Network Parsing the Links and Codes of the Web World.” *Convergence: The International Journal of Research into New Media Technologies* 12 (1): 9–19. doi:10.1177/1354856506061549.

Elmer, Greg, and Ganaele Langlois. 2013. “Networked Campaigns: Traffic Tags and Cross Platform Analysis on the Web.” *Information Polity* 18 (1): 43–56. doi:10.3233/IP-2011-0244.

Evans, David Sparks, Andrei Hagiu, and Richard Schmalensee. 2006. *Invisible Engines: How Software Platforms Drive Innovation and Transform Industries*. MIT Press Cambridge. http://www.hbs.edu/faculty/Pages/print-profile.aspx?facId=337239.

Facebook Developers. n.d. “Canvas.” *Facebook Developers*. https://developers.facebook.com/docs/games/canvas/.

———. 2012. “Open Graph.” *Facebook Developers*. https://developers.facebook.com/docs/opengraph/.

———. 2015a. “A Guide to Sharing for Webmasters.” *Facebook Developers*. Accessed May 26. https://developers.facebook.com/docs/sharing/webmasters.

———. 2014b. “Like Button for iOS Apps.” *Facebook Developers*. Accessed November 11. https://developers.facebook.com/docs/ios/like-button.

———. 2015c. “Open Graph Overview.” *Facebook Developers*. Accessed February 16. https://developers.facebook.com/docs/opengraph/overview/.

———. 2014d. “Platform Policy.” *Facebook Developers*. Accessed November 10. https://developers.facebook.com/policy.

Facebook Help Center. 2015. “Ticker.” Accessed May 25. https://www.facebook.com/help/255898821192992/.

Facebook Investor Relations. 2015. “Facebook Reports First Quarter 2015 Results.” *Investor Relations*. April 22. http://investor.fb.com/releasedetail.cfm?ReleaseID=908022.

Facebook Newsroom. 2006. “Facebook Expands to Include Work Networks.” *Facebook Newsroom*. May 3. https://newsroom.fb.com/news/2006/05/facebook-expands-to-include-work-networks-2/.

———. 2007. “Facebook Unveils Platform for Developers of Social Applications.” *Facebook Newsroom*. May 24. https://newsroom.fb.com/news/2007/05/facebook-unveils-platform-for-developers-of-social-applications/.

Facebook Statistics. 2011. "Facebook Statistics." *Facebook*. https://www.facebook.com/press/info.php?statistics.

"FAQ." 2014. *Internet Archive*. August 8. http://archive.org/about/faqs.php.

Finkelstein, Seth. 2008. "Google, Links, and Popularity versus Authority." In *The Hyperlinked Society*, edited by Joseph Turow and Lokman Tsui, 104–24. Ann Arbor, MI: The University of Michigan Press. http://www.webuse.org/p/c09/.

Foot, Kirsten, and Steven Schneider. 2010. "Object-Oriented Web Historiography." In *Web History*, edited by Niels Brügger, 61–79. New York: Peter Lang. http://faculty.washington.edu/kfoot/Publications/Foot_Schneider.pdf.

Foot, K., S. M. Schneider, M. Dougherty, M. Xenos, and E. Larsen. 2003. "Analyzing Linking Practices: Candidate Sites in the 2002 US Electoral Web Sphere." *Journal of Computer-Mediated Communication* 8 (4): 0–0. http://onlinelibrary.wiley.com/doi/10.1111/j.1083-6101.2003.tb00220.x/full.

Franklin, Brad. 2014. "Facebook's App Links Meets Google's App Indexing. Mobile Deep Linking Is A Game Changer (FB, GOOG)." *Brad Franklin*. July 21. http://bradfranklin.com/2014/07/21/facebooks-app-links-meets-googles-app-indexing-mobile-deep-linking-is-a-game-changer-fb-goog/.

FTC. 2011. "Facebook Settles FTC Charges That It Deceived Consumers By Failing To Keep Privacy Promises." November 29. http://www.ftc.gov/news-events/press-releases/2011/11/facebook-settles-ftc-charges-it-deceived-consumers-failing-keep.

Fuchs, Christian. 2010. "Labor in Informational Capitalism and on the Internet." *The Information Society* 26 (3): 179–96. doi:10.1080/01972241003712215.

Fuller, Matthew. 2003. *Behind the Blip: Essays on the Culture of Software*. Brooklyn, NY: Autonomedia.

———. 2005. *Media Ecologies: Materialist Energies In Art And Technoculture*. Cambridge, MA: The MIT Press.

———. 2006. "Piet Zwart Institute - Software Studies Workshop." *Piet Zwart Institute*. February 17. https://web.archive.org/web/20060217043644/http://pzwart.wdka.hro.nl/mdr/Seminars2/softstudworkshop.

———. 2008. *Software Studies: A Lexicon*. MIT Press.

Funk, Tom. 2008. *Web 2.0 and beyond: Understanding the New Online Business Models, Trends, and Technologies*. Westport, Connecticut: Praeger Publishers.

Galloway, Alexander R. 2006. *Protocol: How Control Exists after Decentralization*. Cambridge, MA: The MIT Press.

Garrett, Sean. 2010. "Links and Twitter: Length Shouldn't Matter." *Twitter Blog*. June 8. http://blog.twitter.com/2010/06/links-and-twitter-length-shouldnt.html.

Gehl, Robert. 2010. "A Cultural and Political Economy of Web 2.0." Fairfax, VA: George Mason University. http://digilib.gmu.edu/dspace/handle/1920/5886.

———. 2011. "The Archive and the Processor: The Internal Logic of Web 2.0." *New Media & Society* 13 (8): 1228–44.

———. 2012. "Real (Software) Abstractions On the Rise of Facebook and the Fall of MySpace." *Social Text* 30 (2 111): 99–119.

———. 2014. *Reverse Engineering Social Media: Software, Culture, and Political Economy in New Media Capitalism*. Philadelphia, Pennsylvania: Temple University Press.

Geiger, Friedrich. 2015. “Facebook Like Button Lands German Sites in Hot Water.” *The Wall Street Journal Blogs - Digits*. May 22. http://blogs.wsj.com/digits/2015/05/22/facebook-like-button-lands-german-sites-in-hot-water/?mod=rss_Technology.

Gelles, David. 2010. “Facebook’s Grand Plan for the Future.” *Financial Times*, December 3. http://www.ft.com/cms/s/2/57933bb8-fcd9-11df-ae2d-00144feab49a.html#axzz1ayaHS1Vs.

Geraci, Anne. 1991. *IEEE Standard Computer Dictionary: Compilation of IEEE Standard Computer Glossaries*. Edited by Freny Katki, Louise McMonegal, Bennett Meyer, John Lane, Paul Wilson, Jane Radatz, Mary Yee, Hugh Porteous, and Fredrick Springsteel. Piscataway, NJ, USA: IEEE Press.

Gerlitz, Carolin, and Anne Helmond. 2013. “The Like Economy: Social Buttons and the Data-Intensive Web.” *New Media & Society* 15 (8): 1348–65. doi:10.1177/1461444812472322.

Ghostery. n.d. “Ghostery™ FAQs.” http://www.ghostery.com/faq.

Gibson, David, Jon Kleinberg, and Prabhakar Raghavan. 1998. “Inferring Web Communities from Link Topology.” In *Proceedings of the Ninth ACM Conference on Hypertext and Hypermedia: Links, Objects, Time and Space—structure in Hypermedia Systems: Links, Objects, Time and Space—structure in Hypermedia Systems*, 225–34. ACM. http://dl.acm.org/citation.cfm?id=276652.

Giles, Jim. 2011. “The Web’s Crystal Ball Gets an Upgrade.” *MIT Technology Review*. November 14. http://www.technologyreview.com/news/426105/the-webs-crystal-ball-gets-an-upgrade/.

Gillespie, Tarleton. 2010. “The Politics of ‘platforms.’” *New Media & Society* 12 (3): 347–64.

———. 2014. “The Relevance of Algorithms.” In *Media Technologies*, edited by Pablo Boczkowski, Kirsten Foot, and Tarleton Gillespie, 167–94. Cambridge, MA: The MIT Press.

———. 2015. “Platforms Intervene.” *Social Media+ Society* 1 (1): 2056305115580479.

Girard, Paul. 2011. “HyperText Corpus Initiative : How to Help Researchers Sieving the Web?” In *Out of the Box: Building and Using Web Archives*. The Hague, the Netherlands. http://www.medialab.sciences-po.fr/publications/Girard-HCI.pdf.

Goddard, Michael. 2011. “Towards an Archaeology of Media Ecologies: ‘Media Ecology’, Political Subjectivation and Free Radios.” *Fibreculture Journal* 17 (Unnatural Ecologies). http://seventeen.fibreculturejournal.org/fcj-114-towards-an-archaeology-of-media-ecologies-%e2%80%98media-ecology%e2%80%99-political-subjectivation-and-free-radios/.

Goddard, Michael, and Jussi Parikka. 2011. “Unnatural Ecologies.” *The Fibreculture Journal* 17. http://seventeen.fibreculturejournal.org/.

Graham, Brad L. 1999. “Friday, September 10, 1999.” *The Bradlands*. September 10. http://www.bradlands.com/weblog/comments/september_10_1999.

Grifantini, Kristina. 2008. “The Future of the Web.” *MIT Technology Review*. June 23. http://www.technologyreview.com/qa/410329/the-future-of-the-web/.

Grosser, Benjamin. n.d. “Facebook Demetricator.” *Benjamin Grosser*. http://bengrosser.com/projects/facebook-demetricator/.

———. 2014. “What Do Metrics Want? How Quantification Prescribes Social Interaction on Facebook.” *Computational Culture*, no. 4 (November).

Grusin, Richard. 2010. *Premediation: Affect and Mediality After 9/11*. Basingstoke, UK: Palgrave Macmillan.

Hagiu, Andrei. 2014. “Strategic Decisions for Multisided Platforms.” *MIT Sloan Management Review*.

Halavais, Alexander. 2008. “The Hyperlink as Organizing Principle.” In *The Hyperlinked Society*, edited by Joseph Turow and Lokman Tsui, 39–55. Ann Arbor, MI: The University of Michigan Press.

Hands, Joss. 2013. “Introduction: Politics, Power and ‘Platformativity.’” *Culture Machine* 14. http://www.culturemachine.net/index.php/cm/article/viewArticle/504.

Hanna, Richard, Andrew Rohm, and Victoria L. Crittenden. 2011. “We’re All Connected: The Power of the Social Media Ecosystem.” *Business Horizons* 54 (3): 265–73.

Hargittai, Eszter. 2008. “The Role of Expertise in Navigating Links of Influence.” In *The Hyperlinked Society*, edited by Joseph Turow and Lokman Tsui, 85–103. Ann Arbor, MI: The University of Michigan Press. http://www.webuse.org/p/c09/.

Hayles, N. Katherine. 2004. “Print Is Flat, Code Is Deep: The Importance of Media-Specific Analysis.” *Poetics Today* 25 (1): 67–90. doi:10.1215/03335372-25-1-67.

Helmond, Anne. 2010. “Identity 2.0: Constructing Identity with Cultural Software.” *New Media Research Blog*. http://www.annehelmond.nl/wordpress/wp-content/uploads/2010/01/helmond_identity20_dmiconference.pdf.

———. 2012. “The Like Economy: The Politics of Data and Dataflows in the Social Web.” In *Proceedings BOBCATSSS 2012 – 20th International Conference on Information Science*, 10–13. Bad Honnef: Bock+Herchen Verlag.

Herring, Susan C., Inna Kouper, John C. Paolillo, Lois Ann Scheidt, Michael Tyworth, Peter Welsch, Elijah Wright, and Ning Yu. 2005. “Conversations in the Blogosphere: An Analysis‘ from the Bottom up.’” In *System Sciences, 2005. HICSS’05. Proceedings of the 38th Annual Hawaii International Conference on*, 107b – 107b. IEEE. http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1385453.

Highfield, Tim. 2009. “Which Way up? Reading and Drawing Maps of the Blogosphere.” *Ejournalist* 9 (1): 99–114.

Hogan, Mél. 2015. “Facebook Data Storage Centers as the Archive’s Underbelly.” *Television & New Media* 16 (1): 3–18. doi:10.1177/1527476413509415.

Holmes, Ryan. 2013. “From Inside Walled Gardens, Social Networks Are Suffocating The Internet As We Know It.” *Fast Company*. August 9. http://www.fastcompany.com/3015418/from-inside-walled-gardens-social-networks-are-suffocating-the-internet-as-we-know-it.

Hourihan, Meg. 2002. “What We’re Doing When We Blog.” *O’Reilly Network*, June. http://www.oreillynet.com/pub/a/javascript/2002/06/13/megnut.html.

Hsu, Chien-leng, and Han Woo Park. 2011. “Sociology of Hyperlink Networks of Web 1.0, Web 2.0, and Twitter: A Case Study of South Korea.” *Social Science Computer Review* 29 (3): 354–68.

Hurst, Matthew. 2007. “Mapping The Blogosphere.” *Data Mining*. http://datamining.typepad.com/gallery/blog-map-gallery.html.

IETF. 2015. “Request for Comments (RFC).” *The Internet Engineering Task Force (IETF®)*. Accessed April 28. https://www.ietf.org/rfc.html.

Ingram, Mathew. 2011. “Blogging Is Dead Just Like the Web Is Dead.” *Gigaom*. February 22. http://gigaom.com/2011/02/22/blogging-is-dead-just-like-the-web-is-dead/.

Instagram. 2012. “Introducing Your Instagram Feed on the Web.” *Instagram Blog*. November 5. http://blog.instagram.com/post/42363074191/instagramfeed.

Ioannides, Nektarios. 2013. “Comment and Like Stuff on Facebook Directly from Bing.” *Bing Blogs. Search Blog*. May 10. http://blogs.bing.com/search/2013/05/10/comment-and-like-stuff-on-facebook-directly-from-bing/.

Jeffrey, Stuart. 2012. “A New Digital Dark Age? Collaborative Web Tools, Social Media and Long-Term Preservation.” *World Archaeology* 44 (4): 553–70. doi:10.1080/00438243.2012.737579.

Jenkins, Henry. 2006. *Convergence Culture: Where Old and New Media Collide*. New York, NY: NYU press.

John, Nicholas A. 2012. “Sharing and Web 2.0: The Emergence of a Keyword.” *New Media & Society*. http://nms.sagepub.com/content/early/2012/06/29/1461444812450684.abstract.

Johnson, Bobbie, Charles Arthur, and Josh Halliday. 2010. “Libyan Domain Shutdown No Threat, Insists Bit.ly.” *The Guardian*, October 9, sec. Technology. http://www.guardian.co.uk/technology/2010/oct/08/bitly-libya.

Kacholia, Varun, and Minwen Ji. 2013. “News Feed FYI: Helping You Find More News to Talk About.” *Facebook Newsroom*. December 2. http://newsroom.fb.com/news/2013/12/news-feed-fyi-helping-you-find-more-news-to-talk-about/.

Kaplan, Andreas M., and Michael Haenlein. 2010. “Users of the World, Unite! The Challenges and Opportunities of Social Media.” *Business Horizons* 53 (1): 59–68. doi:10.1016/j.bushor.2009.09.003.

Karppi, Tero. 2014. “Disconnect. Me. User Engagement and Facebook.” University of Turku. http://www.doria.fi/handle/10024/95616.

Katz, Ian. 2012. “Web Freedom Faces Greatest Threat Ever, Warns Google’s Sergey Brin.” *The Guardian*. April 15. http://www.theguardian.com/technology/2012/apr/15/web-freedom-threat-google-brin.

Kaun, Anne, and Fredrik Stiernstedt. 2014. “Facebook Time: Technological and Institutional Affordances for Media Memories.” *New Media & Society* 16 (7): 1154–68. doi:10.1177/1461444814544001.

Kelly, John, and Bruce Etling. 2008. “Mapping Iran’s Online Public: Politics and Culture in the Persian Blogosphere.” *Berkman Center for Internet and Society and Internet & Democracy Project, Harvard Law School.* http://cyber.law.harvard.edu/sites/cyber.law.harvard.edu/files/Kelly%26Etling_Mapping_Irans_Online_Public_2008.pdf.

Kessler, Sarah. 2014. “Bitly Shortens Its Business Plan.” *Fast Company*. July 18. http://www.fastcompany.com/3033216/innovation-agents/bitly-shortens-its-business-plan.

Kinsey, Mark. 2009. "Keeping Count of Sharing Across the Web." *Facebook Notes*. October 26. https://www.facebook.com/notes/facebook/keeping-count-of-sharing-across-the-web/165161437130.

Kirkpatrick, David. 2010. *The Facebook Effect: The Real Inside Story of Mark Zuckerberg and the World's Fastest Growing Company*. New York: Random House.

Kirkpatrick, Marshall. 2009. "Twitter Crowns Bit.ly As The King of Short Links; Here's What It Means." *ReadWrite*. May 6. http://readwrite.com/2009/05/06/twitter_crowns_bitly_as_the_king_of_short_links_he.

Kirschenbaum, Matthew G. 2003. "Virtuality and Vrml: Software Studies after Manovich." *Electronic Book Review*, no. The Politics of Information. http://www.electronicbookreview.com/thread/technocapitalism/morememory.

———. 2007. *Mechanisms: New Media and the Forensic Imagination*. Cambridge, Mass: The MIT Press.

Kitchin, Rob, and Martin Dodge. 2011. *Code/space: Software and Everyday Life*. Cambridge, MA: MIT Press.

Knowlton, Pete. 2010. "First-Party Tracking vs. Third-Party Tracking." *The Purple Box – The Official Ghostery Blog*. December 28. https://purplebox.ghostery.com/post/1016021484.

Kopytoff, Verne G. 2011. "Blogs Wane as the Young Drift to Sites Like Twitter." *The New York Times*, February 20, sec. Technology / Internet. http://www.nytimes.com/2011/02/21/technology/internet/21blog.html.

Kottke, Jason. 2013. "The Blog Is Dead, Long Live the Blog." *Nieman Journalism Lab*. December 19. http://www.niemanlab.org/2013/12/the-blog-is-dead/.

Krabach, Todd. 2014. "Mobile Like Button Now Available to Everyone." *Facebook Developers*. October 2. https://developers.facebook.com/blog/post/2014/10/02/mobilelike/.

Krim, Tariq. 2007. "Netvibes: (re)mix the Web." presented at the The Next Web Conference, Amsterdam. http://www.slideshare.net/beppe/tariq-krim-at-the-next-web-conference.

Krishnamurthy, Balachander, and Craig Wills. 2009. "Privacy Diffusion on the Web: A Longitudinal Perspective." In *Proceedings of the 18th International Conference on World Wide Web*, 541–50. WWW '09. New York, NY, USA: ACM. doi:10.1145/1526709.1526782.

Kumar, Ravi, Jasmine Novak, Prabhakar Raghavan, and Andrew Tomkins. 2004. "Structure and Evolution of Blogspace." *Communications of the ACM* 47 (12): 35–39.

LaMonica, Martin. 2005. "From Web Page to Web Platform." *CNET*. August 16. http://news.cnet.com/From-Web-page-to-Web-platform/2100-7345_3-5833940.html.

Lane, Kin. 2012. "History of APIs." *API Evangelist*. December 20. http://history.apievangelist.com/.

Langlois, Ganaele. 2005. "Networks and Layers: Technocultural Encodings of the World Wide Web." *Canadian Journal of Communication* 30 (4). http://www.cjc-online.ca/index.php/journal/article/view/1636.

Langlois, Ganaele, and Greg Elmer. 2013. "The Research Politics of Social Media Platforms." *Culture Machine* 14: 1–17.

Langlois, Ganaele, Greg Elmer, Fenwick McKelvey, and Zachary Devereaux. 2009. "Networked Publics: The Double Articulation of Code and Politics on Facebook." *Canadian Journal of Communication* 34 (3). http://www.cjc-online.ca/index.php/journal/article/viewArticle/2114.

Langlois, Ganaele, Fenwick McKelvey, Greg Elmer, and Kenneth Werbin. 2009. "Mapping Commercial Web 2.0 Worlds: Towards a New Critical Ontogenesis." *Fibreculture*, no. 14. http://fourteen.fibreculturejournal.org/fcj-095-mapping-commercial-web-2-0-worlds-towards-a-new-critical-ontogenesis/.

Lardinois, Frederic. 2015. "Facebook Launches AppLinks To Make Deep Linking Between Apps Easier." *TechCrunch*. Accessed May 4. http://social.techcrunch.com/2014/04/30/facebook-launches-applinks-org-to-make-linking-between-apps-easier/.

Latour, Bruno. 1999. *Pandora's Hope: Essays on the Reality of Science Studies*. 1 edition. Cambridge, Mass: Harvard University Press.

———. 2005. "Reassembling the Social-an Introduction to Actor-Network-Theory." *Reassembling the Social-An Introduction to Actor-Network-Theory, by Bruno Latour, Pp. 316. Foreword by Bruno Latour. Oxford University Press, Sep 2005. ISBN-10: 0199256047. ISBN-13: 9780199256044* 1. http://adsabs.harvard.edu/abs/2005reso.book.....L%EF%BF%BD%C3%9C.

Laurent, Simon St, Joe Johnston, Edd Dumbill, and Dave Winer. 2001. *Programming Web Services with XML-RPC*. O'Reilly Media, Inc.

Lazzarato, Maurizio. 1996. "Immaterial Labour." *Radical Thought in Italy: A Potential Politics*, 133–47.

Lee, Sangho, and Jong Kim. 2013. "WarningBird: A Near Real-Time Detection System for Suspicious URLs in Twitter Stream." *IEEE Transactions on Dependable and Secure Computing* 10 (3): 183–95. doi:10.1109/TDSC.2013.3.

Lenhart, Amanda. 2015. "Teens, Social Media & Technology Overview 2015." *Pew Research Center's Internet & American Life Project*. April 9. http://www.pewinternet.org/2015/04/09/teens-social-media-technology-2015/.

Leorke, Dale. 2012. "Rebranding the Platform: The Limitations of 'platform Studies.'" *Digital Culture & Education* 4 (3). http://www.digitalcultureandeducation.com/uncategorized/dce1073_leorke_2012_html/.

Liu, Alan. 2004. "Transcendental Data: Toward a Cultural History and Aesthetics of the New Encoded Discourse." *Critical Inquiry* 31 (1): 49–84.

———. 2008. *Local Transcendence: Essays on Postmodern Historicism and the Database*. Chicago, IL: University of Chicago Press.

Liu, Deborah. 2015. "F8 2015: New Ways to Connect with the Facebook Family of Apps." *Facebook Newsroom*. March 25. https://newsroom.fb.com/news/2015/03/f8-day-one-2015/.

Liu, Hugo. 2007. "Social Network Profiles as Taste Performances." *Journal of Computer-Mediated Communication* 13 (1): 252–75. doi:10.1111/j.1083-6101.2007.00395.x.

Locke, Laura. 2007. "The Future of Facebook." *Time*. July 17. http://content.time.com/time/business/article/0,8599,1644040,00.html.

Lomborg, Stine, and Anja Bechmann. 2014. "Using APIs for Data Collection on Social Media." *The Information Society* 30 (4): 256–65. doi:10.1080/01972243.2014.915276.

Lorenzetti, Laura. 2014. "Instagram Leaves Twitter in the Dust with 300 Million Active Users." *Fortune*. December 10. http://fortune.com/2014/12/10/instagram-leaves-twitter-in-the-dust-with-300-million-active-users/.

Lovink, Geert. 2003. *Dark Fiber: Tracking Critical Internet Culture*. Cambridge: MIT Press.

———. 2008. *Zero Comments: Blogging and Critical Internet Culture*. New York, NY: Routledge.

Lury, Celia, and Liz Moor. 2010. "Blowing up the Brand: Critical Perspectives on Promotional Culture." In *Brand Valuation and Topological Culture*, edited by Melissa Aronczyk and Devon Powers, 21:29–52. New York: Peter Lang. http://eprints.gold.ac.uk/id/eprint/4135.

Mackenzie, Adrian. 2006. *Cutting Code: Software and Sociality*. New York: Peter Lang International Academic Publishers.

MacManus, Richard. 2011. "The Pros & Cons of Frictionless Sharing." *ReadWrite*. September 28. http://readwrite.com/2011/09/28/frictionless_sharing_pros_cons.

Madden, Mary, and Susannah Fox. 2006. "Riding the Waves of 'Web 2.0.'" *Pew Internet and American Life Project* 5. http://www.pewinternet.org/files/old-media/Files/Reports/2006/PIP_Web_2.0.pdf.pdf.

Madrigal, Alexis C. 2012. "Dark Social: We Have the Whole History of the Web Wrong." *The Atlantic*. October 12. http://www.theatlantic.com/technology/archive/2012/10/dark-social-we-have-the-whole-history-of-the-web-wrong/263523/.

Maggi, Federico, Alessandro Frossi, Stefano Zanero, Gianluca Stringhini, Brett Stone-Gross, Christopher Kruegel, and Giovanni Vigna. 2013. "Two Years of Short URLs Internet Measurement: Security Threats and Countermeasures." In *Proceedings of the 22nd International Conference on World Wide Web*, 861–72. WWW '13. Republic and Canton of Geneva, Switzerland: International World Wide Web Conferences Steering Committee. http://dl.acm.org/citation.cfm?id=2488388.2488463.

Makino, Takaki, Chaesang Jung, and Doantam Phan. 2015. "Finding More Mobile-Friendly Search Results." *Official Google Webmaster Central Blog*. February 26. http://googlewebmastercentral.blogspot.com/2015/02/finding-more-mobile-friendly-search.html.

Malik, Om. 2006. "Suddenly Everything's Coming up Widgets." *CNN Money*. October 2. http://money.cnn.com/magazines/business2/business2_archive/2006/09/01/8384338/index.htm.

Manovich, Lev. 2001. *The Language of New Media*. 1st MIT Press pbk. ed edition. Cambridge, MA: The MIT Press.

———. 2013. *Software Takes Command*. New York, NY: Bloomsbury Academic.

Marino, Mark C. 2006. "Critical Code Studies." *Electronic Book Review*, December. http://www.electronicbookreview.com/thread/electropoetics/codology.

———. 2014. "Field Report for Critical Code Studies 2014." *Computational Culture*, no. 4 (November).

Marres, Noortje, and Esther Weltevrede. 2013. "Scraping the Social?" *Journal of Cultural Economy* 6 (3): 313–35. doi:10.1080/17530350.2013.772070.

Marshall, Matt. 2007. "Friendfeed, a Company Lets You See All Your Friends' Feeds." *VentureBeat*. October 1. http://venturebeat.com/2007/10/01/friendfeed-a-company-lets-you-see-all-your-friends-feeds/.

Mattlemay. 2011. "Bitly and Verisign." *Bitly Blog*. November 3. http://blog.bitly.com/post/12296694165/bitly-and-verisign.

Mayer, Jonathan R., and John C. Mitchell. 2012. "Third-Party Web Tracking: Policy and Technology." In *Security and Privacy (SP), 2012 IEEE Symposium on*, 413–27. http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6234427.

Mayer, Marissa. 2013. "@yahoo Delivers #bestoftheweb." *Yahoo! Yodel*. May 16. https://yodel.yahoo.com/blogs/general/yahoo-delivers-bestoftheweb-160346039.html.

McCarthy, Caroline. 2010. "Why the Social-Media Aggregator Has Croaked." *CNET*. March 3. http://www.cnet.com/news/why-the-social-media-aggregator-has-croaked/.

McCown, Frank, and Michael L. Nelson. 2009. "What Happens When Facebook Is Gone?" In *Proceedings of the 9th ACM/IEEE-CS Joint Conference on Digital Libraries*, 251–54. JCDL '09. New York, NY, USA: ACM. doi:10.1145/1555400.1555440.

McKelvey, Fenwick. 2011. "A Programmable Platform? Drupal, Modularity, and the Future of the Web." *Fibreculture* 18. http://eighteen.fibreculturejournal.org/2011/10/09/fcj-128-programmable-platform-drupal-modularity-and-the-future-of-the-web/.

Media on Facebook. 2010. "The Value of a Liker." *Facebook Notes*. September 29. https://www.facebook.com/note.php?note_id=150630338305797.

Meeuwsen, Frank. 2010. *Bloghelden*. Utrecht: A.W. Bruna Uitgevers B.V.

Megiddo, Nimrod, and Kevin S. McCurley. 2005. Efficient retrieval of uniform resource locators. 6957224, issued October 18, 2005. http://www.google.com/patents/US6957224.

Metz, Cade. 2013. "Facebook Buys Its Way Into the Heart of the App World." *Wired*. September 12. http://www.wired.com/2013/09/facebook-and-parse/.

Montfort, Nick, and Ian Bogost. 2009. *Racing the Beam: The Atari Video Computer System*. 2nd edition. Cambridge, Mass: The MIT Press.

Morin, Dave. 2008. "Announcing Facebook Connect." *Facebook Developers Blog*. May 9. https://developers.facebook.com/blog/post/2008/05/09/announcing-facebook-connect/.

Morozov, Evgeny. 2011. "The Internet Intellectual." *The New Republic*, October 12. http://www.newrepublic.com/article/books/magazine/96116/the-internet-intellectual.

Moschella, David C. 1997. *Waves of Power: The Dynamics of Global Technology Leadership, 1964-2010*. New York: AMACOM.

Mott, Nathaniel. 2013. "Destroying the App Store Hegemony: Twitter's New Cards Could Destroy App Silos." *PandoDaily*. April 3. http://pando.com/2013/04/03/destroying-the-app-store-hegemony-twitters-new-cards-could-destroy-app-silos/.

Murugesan, San. 2007. "Understanding Web 2.0." *IT Professional* 9 (4): 34–41. doi:10.1109/MITP.2007.78.

Myllymaki, Jussi. 2002. "Effective Web Data Extraction with Standard XML Technologies." *Computer Networks* 39 (5): 635–44.

Naaijkens, Maarten. 2013. "Hyves Transformeert Naar Online Gaming Portal." Amsterdam, Netherlands. http://hyves-corporate.pr.co/63596-hyves-transformeert-naar-online-gaming-portal.

Neumann, Alexander, Johannes Barnickel, and Ulrike Meyer. 2010. "Security and Privacy Implications of Url Shortening Services." In *Proceedings of the Workshop on Web 2.0 Security and Privacy*. http://w2spconf.com/2011/papers/w2sp-urlShortening-slides.pdf.

Niederer, Sabine, and José Van Dijck. 2010. "Wisdom of the Crowd or Technicity of Content? Wikipedia as a Sociotechnical System." *New Media & Society* 12 (8): 1368–87. doi:10.1177/1461444810365297.

Nisenholtz, Martin. 2008. "The Hyperlinked News Organization." In *The Hyperlinked Society*, edited by Joseph Turow and Lokman Tsui, 104–24. Ann Arbor, MI: The University of Michigan Press. http://www.webuse.org/p/c09/.

Oliver, Julian, Gordan Savičić, and Danja Vasiliev. 2011. "The Critical Engineering Manifesto." *The Critical Engineering Working Group*. http://criticalengineering.org/.

Olsen, Stefanie. 2006. "Social Networks--Future Portal or Fad?" *CNET*. June 14. http://news.cnet.com/2100-1038_3-6083546.html.

O'Reilly, Tim. 2005. "What Is Web 2.0: Design Patterns and Business Models for the next Generation of Software." *O'Reilly*. September 30. http://oreilly.com/web2/archive/what-is-web-20.html.

O'Reilly, Tim, and John Battelle. 2004. "Opening Welcome: The State of the Internet Industry." presented at the Web 2.0 Conference, Hotel Nikko, San Francisco, CA, October 5. http://web2con.com/presentations/web2con/intro_tim_john.ppt.

Orr, Caitlin R., Arun Chauhan, Minaxi Gupta, Christopher J. Frisz, and Christopher W. Dunn. 2012. "An Approach for Identifying JavaScript-Loaded Advertisements Through Static Program Analysis." In *Proceedings of the 2012 ACM Workshop on Privacy in the Electronic Society*, 1–12. WPES '12. New York, NY, USA: ACM. doi:10.1145/2381966.2381968.

Page, Lawrence, Sergey Brin, Rajeev Motwani, and Terry Winograd. 1999. "The PageRank Citation Ranking: Bringing Order to the Web." http://ilpubs.stanford.edu:8090/422.

Papacharissi, Zizi, ed. 2010. "Conclusion: A Networked Self." In *A Networked Self: Identity, Community, and Culture on Social Network Sites*, 304–17. New York: Routledge.

Parikka, Jussi. 2011. "Platform Politics-Conference: Opening Words." *Machinology*. May 15. http://jussiparikka.net/2011/05/15/platform-politics-conference-opening-words/.

Park, Han Woo. 2003. "Hyperlink Network Analysis: A New Method for the Study of Social Structure on the Web." *Connections* 25: 49–61.

Patelis, Korinna. 2000. "E-Mediation by America Online." In *Preferred Placement: Knowledge Politics on the Web*, edited by Richard Rogers, 49–63. Maastricht: Jan van Eyck Akademie Editions.

Pearlman, Leah. 2009. "I like This." *Facebook Notes*. February 10. https://www.facebook.com/notes/facebook/i-like-this/53024537130.

Petersen, Søren Mørk. 2008. "Loser Generated Content: From Participation to Exploitation." *First Monday* 13 (3). http://firstmonday.org/ojs/index.php/fm/article/view/2141.

Peters, Thomas A. 1999. *Computerized Monitoring and Online Privacy*. McFarland.

Poell, Thomas. 2013. “Social Media and the Transformation of Activist Communication: Exploring the Social Media Ecology of the 2010 Toronto G20 Protests.” *Information, Communication & Society* 17 (6): 716–31. doi:10.1080/1369118X.2013.812674.

Postman, Neil. 1970. “The Reformed English Curriculum.” In *High School 1980: The Shape of the Future in American Secondary Education*, edited by A.C. Eurich, 160–68. New York: Pitman.

Programmable Web. n.d. “Frequently Asked Questions.” *Programmable Web*.

Protalinski, Emil. 2015. “Facebook Passes 1.44B Monthly Active Users and 1.25B Mobile Users; 65% Are Now Daily Users.” *VentureBeat*. April 22. http://venturebeat.com/2015/04/22/facebook-passes-1-44b-monthly-active-users-1-25b-mobile-users-and-936-million-daily-users/.

Puschmann, Cornelius, and Jean Burgess. 2013. “The Politics of Twitter Data.” In *HIIG Discussion Paper Series No. 2013-01*. Social Science Research Network. http://papers.ssrn.com/abstract=2206225.

Quick, William. 2002. *Daily Pundit*. January 1. http://web.archive.org/web/20020611194347/http://www.iw3p.com/DailyPundit/2001_12_30_dailypundit_archive.html.

Quiggin, John. 2013. “The Economics of New Media.” In *A Companion to New Media Dynamics*, edited by John Hartley, Jean Burgess, and Axel Bruns, 90–103. West Sussex: John Wiley & Sons.

Radwanick, Sarah. 2012. “Facebook Continues Its Global Dominance, Claiming the Lead in Brazil.” *comScore, Inc*. January 20. http://www.comscore.com/Insights/Blog/Facebook-Continues-its-Global-Dominance-Claiming-the-Lead-in-Brazil.

Renzi, Alessandra. 2011. “What Is the Politics of Platform Politics?” *Television & New Media* 12 (5): 483–85.

Richardson, Leonard, and Sam Ruby. 2008. *RESTful Web Services*. Sebastopol, CA: O’Reilly Media, Inc.

Rieder, Bernhard. 2013. “Scrutinizing a Network of Likes on Facebook (and Some Thoughts on Network Analysis and Visualization).” *The Politics of Systems*. July 10. http://thepoliticsofsystems.net/2013/07/scrutinizing-a-network-of-likes-on-facebook-and-some-thoughts-on-network-analysis-and-visualization/.

Rieder, Bernhard, and Guillaume Sire. 2013. “Conflicts of Interest and Incentives to Bias: A Microeconomic Critique of Google’s Tangled Position on the Web.” *New Media & Society*, 1461444813481195.

Ritzer, George, and Nathan Jurgenson. 2010. “Production, Consumption, Prosumption The Nature of Capitalism in the Age of the Digital ‘prosumer.’” *Journal of Consumer Culture* 10 (1): 13–36. doi:10.1177/1469540509354673.

Rochet, Jean-Charles, and Jean Tirole. 2003. “Platform Competition in Two-Sided Markets.” *Journal of the European Economic Association* 1 (4): 990–1029. doi:10.1162/154247603322493212.

Roesner, Franziska, Tadayoshi Kohno, and David Wetherall. 2012. “Detecting and Defending Against Third-Party Tracking on the Web.” In *Proceedings of the 9th USENIX Conference on Networked Systems Design and Implementation*, 12–12. NSDI’12. Berkeley, CA, USA: USENIX Association. http://dl.acm.org/citation.cfm?id=2228298.2228315.

Rogers, Richard. 2002. “Operating Issue Networks on the Web.” *Science as Culture* 11 (2): 191–213.

———. 2005. “Old and New Media: Competition and Political Space.” *Theory & Event* 8 (2). http://muse.jhu.edu/journals/theory_and_event/v008/8.2rogers.html.

———. 2006. *Information Politics on the Web*. Cambridge, MA: The MIT Press.

———. 2013. *Digital Methods*. Cambridge, MA: The MIT Press.

Rogers, Richard, and Anat Ben-David. 2008. “The Palestinian-Israeli Peace Process and Transnational Issue Networks: The Complicated Place of the Israeli NGO.” *New Media & Society* 10 (3): 497–528. doi:10.1177/1461444807085321.

Roosendaal, Arnold. 2010. “Facebook Tracks and Traces Everyone: Like This!” SSRN Scholarly Paper ID 1717563. Rochester, NY: Social Science Research Network. http://papers.ssrn.com/abstract=1717563.

Rose, Kevin. 2006. “Integrating Digg Within Your Website.” *Digg Blog*. July. http://diggtheblog.blogspot.nl/2006/07/integrating-digg-within-your-website.html.

Rosenberg, Scott. 2010. *Say Everything: How Blogging Began, What It’s Becoming, and Why It Matters*. New York: Random House LLC.

Ross, Joshua-Michéle. 2009. “John Hagel on The Social Web.” *O’Reilly Radar*. October 24. http://radar.oreilly.com/2009/10/john-hagel-on-the-social-web.html.

Rothbart, Andrew. 2012. “Bringing Like to Mobile.” *Facebook Developers*. June 20. https://developers.facebook.com/blog/post/2012/06/20/bringing-like-to-mobile/.

Rowinski, Dan. 2013. “Facebook Hijacks Internet Sites For An Hour Thursday Afternoon.” *ReadWrite*. February 8. http://readwrite.com/2013/02/08/facebook-hijacks-internet-sites-for-an-hour.

Ruppert, Evelyn, John Law, and Mike Savage. 2013. “Reassembling Social Science Methods: The Challenge of Digital Devices.” *Theory, Culture & Society* 30 (4): 22–46. doi:10.1177/0263276413484941.

SalahEldeen, Hany M., and Michael L. Nelson. 2012. “Losing My Revolution: How Many Resources Shared on Social Media Have Been Lost?” In *Theory and Practice of Digital Libraries*, edited by Panayiotis Zaphiris, George Buchanan, Edie Rasmussen, and Fernando Loizides, 125–37. Lecture Notes in Computer Science 7489. Springer Berlin Heidelberg. http://link.springer.com/chapter/10.1007/978-3-642-33290-6_14.

Salter, Anastasia, and John Murray. 2014. *Flash: Building the Interactive Web*. Platform Studies 4. Cambridge, Massachusetts: The MIT Press.

Samar, Thaer, Hugo Huurdeman, Anat Ben-David, Jaap Kamps, and Arjen de Vries. 2014. “Uncovering the Unarchived Web.” In *Proceedings of the 37th International ACM SIGIR Conference on Research &#38; Development in Information Retrieval*, 1199–1202. SIGIR ’14. New York, NY, USA: ACM. doi:10.1145/2600428.2609544.

Samson, Ted. 2013. “Facebook Error That Hijacks Thousands of Websites Isn’t Just an ‘Inconvenience.’” *InfoWorld*. February 8. http://www.infoworld.com/article/2613450/internet-privacy/facebook-error-that-hijacks-thousands-of-websites-isn-t-just-an--inconvenience-.html.

Sandvig, Christian. 2013. “The Internet as Infrastructure.” In *The Oxford Handbook of Internet Studies*, edited by W. H. Dutton, 607. Oxford, UK: Oxford University Press.

Sawers, Paul. 2010. "Facebook Dislike Button: Why It Will Never Happen." *The Next Web*. October 10. http://thenextweb.com/socialmedia/2010/10/10/facebook-dislike-button-why-it-will-never-happen/.

Schaap, Frank. 2004. "Links, Lives, Logs: Presentation in the Dutch Blogosphere." Edited by Laura Gurak, Smiljana Antonijevic, Laurie Johnson, Clancy Ratliff, and Jessica Reyman. *Into the Blogosphere: Rhetoric, Community, and Culture of Weblogs*. http://blog.lib.umn.edu/blogosphere/links_lives_logs.html.

Schachter, Joshua. 2009. "On Url Shorteners." *Joshua Schachter's Blog*. April 3. http://joshua.schachter.org/2009/04/on-url-shorteners.html.

Schäfer, Mirko Tobias. 2011. *Bastard Culture!: How User Participation Transforms Cultural Production*. Amsterdam: Amsterdam University Press.

Schmidt, Jürgen. 2011. "2 Klicks Für Mehr Datenschutz." *Heise Online*. September 1. http://www.heise.de/ct/artikel/2-Klicks-fuer-mehr-Datenschutz-1333879.html.

Schneider, S. M., and K. A. Foot. 2004. "The Web as an Object of Study." *New Media and Society* 6: 114–22.

———. 2010. "Object Oriented Web Historiography." *Web History. New York: Peter Lang Publishing*.

Scholz, T., and P. Hartzog. 2007. "Toward a Critique of the Social Web." *Re-Public: Reimagining Democracy*. https://web.archive.org/web/20070423071707/http://www.re-public.gr/en/?p=201.

Schonfeld, Erick. 2010. "Zuckerberg: 'We Are Building A Web Where The Default Is Social.'" *Techcrunch*. April 21. http://techcrunch.com/2010/04/21/zuckerbergs-buildin-web-default-social/.

Schrems, Max. 2011. "Complaint against Facebook Ireland Ltd. – 02 'Shadow Profiles.'" http://europe-v-facebook.org/Compalint_02_Shadow_Profiles.pdf.

Scolari, Carlos A. 2012. "Media Ecology: Exploring the Metaphor to Expand the Theory." *Communication Theory* 22 (2): 204–25.

Shaw, Aaron, and Yochai Benkler. 2012. "A Tale of Two Blogospheres Discursive Practices on the Left and Right." *American Behavioral Scientist* 56 (4): 459–87.

Siles, I. 2011. "From Online Filter to Web Format: Articulating Materiality and Meaning in the Early History of Blogs." *Social Studies of Science* 41 (5): 737–58.

Singh, Harshdeep. 2014. "Graph API v2.1 and Updated iOS and Android SDKs." *Facebook Developers*. August 7. https://developers.facebook.com/blog/post/2014/08/07/Graph-API-v2.1/.

Singletary, Taylor. 2011. "Twitter's T.co URL Wrapper Is Now on for All URLs 19 Characters and Greater." *Twitter Developers Discussions*. August. https://dev.twitter.com/discussions/1062.

Song, Felicia Wu. 2010. "Theorizing Web 2.0: A Cultural Perspective." *Information, Communication & Society* 13 (2): 249–75.

Sotomayor, Bernardo Rafael. 1998. Automatic summary page creation and hyperlink generation. 5708825, filed May 26, 1995, and issued January 13, 1998. http://www.google.com/patents/US5708825.

Stalder, Felix. 2012. "Between Democracy and Spectacle: The Front-End and Back-End of the Social Web." In *The Social Media Reader*, edited by M. Mandiberg, 242–56. New York: New York University Press.

Star, Susan Leigh, and Karen Ruhleder. 1996. "Steps Toward an Ecology of Infrastructure: Design and Access for Large Information Spaces." *Information Systems Research* 7 (1): 111–34.

Steinmueller, William Edward. 1996. "The U.S. Software Industry: An Analysis and Interpretive History." In *The International Computer Software Industry: A Comparative Study of Industry Evolution and Structure*, edited by David C. Mowery, 15–52. New York: Oxford University Press.

Sterling, Greg. 2009. "URL Shorteners Come To Google & Facebook." *Search Engine Land*. December 14. http://searchengineland.com/url-shorteners-come-to-google-facebook-31880.

Sterne, Jonathan. 2014. "'What Do We Want?''Materiality!''When Do We Want It?''Now!'" In *Media Technologies: Essays on Communication, Materiality, and Society*, edited by Tarleton Gillespie, Pablo Boczkowski, and Kirsten Foot, 119–28. Cambridge: MIT Press.

Stevenson, Michael. 2010a. "The Archived Blogosphere: Exploring Web Historical Methods Using the Internet Archive." In *Digital Methods Mini–conference, University of Amsterdam (January)*.

———. 2010b. "The Archived Blogosphere: Exploring Web Historical Methods Using the Internet Archive." In . University of Amsterdam.

———. 2013. "The Web as Exception: The Rise of New Media Publishing Cultures." Amsterdam: University of Amsterdam. http://dare.uva.nl/document/485746.

———. 2014. "Rethinking the Participatory Web: A History of HotWired's 'new Publishing Paradigm,' 1994–1997." *New Media & Society*, October, 1461444814555950. doi:10.1177/1461444814555950.

Stross, Randall. 2009. *Planet Google: One Company's Audacious Plan to Organize Everything We Know*. Simon and Schuster.

Sukhar, Ilya. 2014. "f8: Helping Developers Build, Grow, and Monetize." *Facebook Developers*. April 30. https://developers.facebook.com/blog/post/2014/04/30/f8/.

Sullivan, Danny. 2011. "As Deal With Twitter Expires, Google Realtime Search Goes Offline." *Search Engine Land*. July 4. http://searchengineland.com/as-deal-with-twitter-expires-google-realtime-search-goes-offline-84175.

———. 2015. "The Google-Twitter Deal Goes Live, Giving Tweets Prominent Placement In Google's Results." *Search Engine Land*. May 19. http://searchengineland.com/google-twitter-deal-live-221148.

Taylor, Bret. 2007. "I like It, I like It." *FriendFeed Blog*. October 30. http://blog.friendfeed.com/2007/10/i-like-it-i-like-it.html.

———. 2009. "FriendFeed Accepts Facebook Friend Request." *FriendFeed Blog*. August 10. http://blog.friendfeed.com/2009/08/friendfeed-accepts-facebook-friend.html.

Terranova, Tiziana. 2000. "Free Labor: Producing Culture for the Digital Economy." *Social Text* 18 (2): 33–58.

The Bing Team. 2013. "Twitter and Bing." *Bing Blogs. Search Blog*. November 1. http://blogs.bing.com/search/2013/11/01/twitter-and-bing/.

Thrift, Nigel. 2005. *Knowing Capitalism*. London, UK: Sage Publications Ltd.

Tokuda, Lance. 2009. "Social Applications: Growth, Use, and Monetization." presented at the Global ICT Summit 2009, Tokyo, June 8. http://www.ict-summit.jp/2009/pdf/report6_tokuda.pdf.

Tufekci, Zeynep. 2008. "Can You See Me Now? Audience and Disclosure Regulation in Online Social Network Sites." *Bulletin of Science, Technology & Society* 28 (1): 20–36. doi:10.1177/0270467607311484.

Turow, Joseph. 2008. "Introduction: On Not Taking the Hyperlink for Granted." In *The Hyperlinked Society: Questioning Connections in the Digital Age*, edited by Joseph Turow and Lokman Tsui. Ann Arbor, MI: University of Michigan Press.

Twitter. 2011. "Link Sharing Made Simple." *Twitter Blog*. June 7. https://blog.twitter.com/2011/link-sharing-made-simple.

———. 2012a. "Discover: Improved Personalization Algorithms and Real-Time Indexing." *Twitter Blog*. May 1. https://blog.twitter.com/2012/discover-improved-personalization-algorithms-and-real-time-indexing.

———. 2012b. "Twitter Privacy Policy." May 17. https://twitter.com/privacy.

Ullrich, Carsten, Kerstin Borau, Heng Luo, Xiaohong Tan, Liping Shen, and Ruimin Shen. 2008. "Why Web 2.0 Is Good for Learning and for Research: Principles and Prototypes." In *Proceedings of the 17th International Conference on World Wide Web*, 705–14. WWW '08. New York, NY, USA: ACM. doi:10.1145/1367497.1367593.

Unabhängiges Landeszentrum für Datenschutz Schleswig-Holstein. 2011a. "Facebook-Fanpages Und -Plugins: Das ULD Weiß, Was Es Tut." *Unabhängiges Landeszentrum Für Datenschutz Schleswig-Holstein*. July 10. https://www.datenschutzzentrum.de/presse/20111007-facebook-fanpages.htm.

———. 2011b. "ULD an Webseitenbetreiber: 'Facebook-Reichweitenanalyse Abschalten.'" *Unabhängiges Landeszentrum Für Datenschutz Schleswig-Holstein*. August 19. https://www.datenschutzzentrum.de/presse/20110819-facebook.htm.

———. 2012. "ULD Issues Orders against Facebook because of Mandatory Real Names." *Unabhängiges Landeszentrum Für Datenschutz Schleswig-Holstein*. December 17. https://www.datenschutzzentrum.de/presse/20121217-facebook-real-names.htm.

———. 2014. "Verwaltungsrechtssache Unabhängiges Landeszentrum Für Datenschutz ./. Wirtschaftsakademie Schleswig-Holstein GmbH 4 LB 20/13, Ihr Schreiben Vom 16. Juni 2014." *Unabhängiges Landeszentrum Für Datenschutz Schleswig-Holstein*. June 16. https://www.datenschutzzentrum.de/facebook/20140721-erwiderung.html.

Van den Boomen, Marianne. 2014. *Transcoding the Digital: How Metaphors Matter in New Media*. Theory on Demand 14. Amsterdam: Institute of Network Cultures. http://networkcultures.org/wpmu/portal/publication/no-14-transcoding-the-digital/.

Van Dijck, José. 2009. "Users like You? Theorizing Agency in User-Generated Content." *Media, Culture, and Society* 31 (1). http://mcs.sagepub.com/content/31/1/41.full.pdf.

———. 2013a. "Engineering Sociality in a Culture of Connectivity." In *The Culture of Connectivity: A Critical History of Social Media*, 3–23. New York: Oxford University Press.

———. 2013b. "Facebook and the Imperative of Sharing." In *The Culture of Connectivity: A Critical History of Social Media*. New York, NY: Oxford University Press.

———. 2013c. *The Culture of Connectivity: A Critical History of Social Media*. New York: Oxford University Press.

Van Dijck, José, and David Nieborg. 2009. "Wikinomics and Its Discontents: A Critical Analysis of Web 2.0 Business Manifestos." *New Media & Society* 11 (5): 855–74. doi:10.1177/1461444809105356.

Vis, Farida. 2013. "A Critical Reflection on Big Data: Considering APIs, Researchers and Tools as Data Makers." *First Monday* 18 (10). http://firstmonday.org/ojs/index.php/fm/article/view/4878.

Vogelstein, F. 2009. "Great Wall of Facebook: The Social Network's Plan to Dominate the Internet—and Keep Google Out." *Wired Magazine* 17 (10): 96–121.

W3C. 2013a. "Links." Accessed July 17. http://www.w3.org/TR/html401/struct/links.html.

———. 2014b. "XML ESSENTIALS." *W3C*. Accessed May 28. http://www.w3.org/standards/xml/core.

Walker, Jill. 2002. "Links and Power: The Political Economy of Linking on the Web." In *Proceedings of the Thirteenth ACM Conference on Hypertext and Hypermedia*, 72–73. http://dl.acm.org/citation.cfm?id=513358.

Walker Rettberg, Jill. 2013. *Blogging*. 2nd Edition edition. Cambridge: Polity Press.

WatchMouse. 2010. "URL Shorteners Make the Web Substantially Slower; Facebooks' Fb.me Is Slowest." *The Official WatchMouse Blog*. March. http://blog.watchmouse.com/2010/03/url-shorteners-make-the-web-substantially-slower-facebooks-fb-me-is-slowest/.

Watkins, S. Craig. 2010. *The Young and the Digital: What the Migration to Social Network Sites, Games, and Anytime, Anywhere Media Means for Our Future*. Boston: Beacon Press.

Wayner, Peter. 2013. "How Facebook Connect Took down the Web." *InfoWorld*. February 12. http://www.infoworld.com/article/2613508/application-development/how-facebook-connect-took-down-the-web.html.

Weltevrede, Esther. 2009. "Thinking Nationally with the Web: A Medium-Specific Approach to the National Turn in Web Archiving." Amsterdam: University of Amsterdam.

———. 2011. "Discussion during the Digital Methods Initiative Winter Conference 2011." University of Amsterdam.

Weltevrede, Esther, and Anne Helmond. 2012. "Where Do Bloggers Blog? Platform Transitions within the Historical Dutch Blogosphere." *First Monday* 17 (2). doi:10.5210/fm.v17i2.3775.

Whitelaw, Mitchell. 2008. "Art Against Information: Case Studies in Data Practice." *The Fibreculture Journal*, no. 11. http://eleven.fibreculturejournal.org/fcj-067-art-against-information-case-studies-in-data-practice/.

Wikipedia. 2008. "Closed Platform." *Wikipedia, the Free Encyclopedia*. http://en.wikipedia.org/w/index.php?title=Closed_platform&oldid=649488745.

Wilhelm, Alex. 2013. "Apparent Issue with Facebook Connect Is Dragging People from around the Web to a Moot Error Page." *The Next Web*. February 8. http://thenextweb.com/facebook/2013/02/08/apparent-issue-with-facebook-connect-is-dragging-people-from-around-the-web-to-a-moot-error-page/.

Willmott, Steven, and Guillaume Balas. n.d. *Winning in the API Economy*.

Winer, Dave. 1995. "What Is a Platform?" *DaveNet*. August 22. http://scripting.com/davenet/1995/08/22/whatisaplatform.html.

———. 2003. “What Makes a Weblog a Weblog?” *Weblogs At Harvard Law*. May 23. http://blogs.law.harvard.edu/whatmakesaweblogaweblog.html.

———. 2009. “Solving the TinyUrl Centralization Problem.” *Scripting News*. March 7. http://scripting.com/stories/2009/03/07/solvingTheTinyurlCentraliz.html.

———. 2011. “This Is a Facebook-Free Zone.” *Scripting News*. December 24. http://scripting.com/stories/2011/12/24/facebookfreeZone.html.

Wu, Tim. 2011. *The Master Switch: The Rise and Fall of Information Empires*. Reprint edition. New York: Vintage.

YouTube. 2005. “August 2005.” *YouTube Official Blog*. August 21. http://youtube-global.blogspot.nl/2005_08_01_archive.html.

Yu, Shuli, and C. Jason Woodard. 2009. “Innovation in the Programmable Web: Characterizing the Mashup Ecosystem.” In *Service-Oriented Computing–ICSOC 2008 Workshops*, 136–47. Springer. http://link.springer.com/chapter/10.1007/978-3-642-01247-1_13.

Zickuhr, Kathryn. 2010. “Generations 2010.” *Pew Research Center’s Internet & American Life Project*. December 16. http://www.pewinternet.org/2010/12/16/generations-2010/.

Zittrain, Jonathan. 2009. *The Future of the Internet--And How to Stop It*. New Haven, Conn.: Yale University Press.

Zuckerberg, Mark. 2010. “Building the Social Web Together.” *Facebook Notes*. April 21. https://www.facebook.com/notes/facebook/building-the-social-web-together/383404517130.

———. 2012. “One Billion People on Facebook.” *Facebook Newsroom*. October 4. http://newsroom.fb.com/news/2012/10/one-billion-people-on-facebook/.

Appendices

Appendix A: GREL coded actors in the Dutch blogosphere (I)

```
{
    "op": "core/text-transform",
    "description": "Text transform on cells in column Platforms using expression
grel:value.replace(/.*blogger.com\\/profile.*/,\"blogger.com/profile\")",
    "engineConfig": {
      "facets": [],
      "mode": "row-based"
    },
    "columnName": "Platforms",
    "expression": "grel:value.replace(/.*blogger.com\\/profile.*/,\"blogger.com/profile\")",
    "onError": "set-to-blank",
    "repeat": false,
    "repeatCount": 10
  },
  {
    "op": "core/text-transform",
    "description": "Text transform on cells in column Platforms using expression
grel:value.replace(/.*.worldonline.nl\\/~.*/,\"worldonline\")",
    "engineConfig": {
      "facets": [],
      "mode": "row-based"
    },
    "columnName": "Platforms",
    "expression": "grel:value.replace(/.*.worldonline.nl\\/~.*/,\"worldonline\")",
    "onError": "set-to-blank",
    "repeat": false,
    "repeatCount": 10
  },
 {
    "op": "core/text-transform",
    "description": "Text transform on cells in column Platforms using expression
grel:value.replace(/.*members.aol.com\\/.*/,\"aol\")",
    "engineConfig": {
      "facets": [],
      "mode": "row-based"
    },
    "columnName": "Platforms",
    "expression": "grel:value.replace(/.*members.aol.com\\/.*/,\"aol\")",
    "onError": "set-to-blank",
    "repeat": false,
    "repeatCount": 10
  },
```

Appendix B: GREL coded actors in the Dutch blogosphere (II)

```
[
  {
    "op": "core/text-transform",
    "description": "Text transform on cells in column Platforms using expression grel:if(
value.contains(\"facebook.com/pages\"),
value.replace(/.*facebook.com\\/pages.*/,\"facebook_pages\"), if(
value.contains(\"facebook.com/group\"),
value.replace(/.*facebook.com\\/group.*/,\"facebook_group\"), if(
value.contains(\"facebook.com/people\"),
value.replace(/.*facebook.com\\/people.*/,\"facebook_people\"), if(
value.contains(\"facebook.com/profile\"),
value.replace(/.*facebook.com\\/profile.*/,\"facebook_profile\"), if(
value.contains(\"facebook.com\"), value.replace(/.*facebook.com.*/,value), value ) ) ) ) )",
    "engineConfig": {
      "facets": [
        {
          "query": "facebook",
          "name": "Platforms",
          "caseSensitive": false,
          "columnName": "Platforms",
          "type": "text",
          "mode": "text"
        }
      ],
      "mode": "row-based"
    },
    "columnName": "Platforms",
    "expression": "grel:if( value.contains(\"facebook.com/pages\"),
value.replace(/.*facebook.com\\/pages.*/,\"facebook_pages\"), if(
value.contains(\"facebook.com/group\"),
value.replace(/.*facebook.com\\/group.*/,\"facebook_group\"), if(
value.contains(\"facebook.com/people\"),
value.replace(/.*facebook.com\\/people.*/,\"facebook_people\"), if(
value.contains(\"facebook.com/profile\"),
value.replace(/.*facebook.com\\/profile.*/,\"facebook_profile\"), if(
value.contains(\"facebook.com\"), value.replace(/.*facebook.com.*/,value), value ) ) ) ) )",
    "onError": "set-to-blank",
    "repeat": false,
    "repeatCount": 10
  },]
```

Appendix C: URL Follow with cURL

The URL Follow tool has been custom built by Bernhard Rieder for this research: http://labs.polsys.net/tools/urlfollow/ [Accessed 1 June 2014]. The tool resolves the URL redirection paths of a single URL using cURL. The input is a single shortened URL and the output is the HTTP header for each redirection. The location of the redirect is displayed in bold.

```
INPUT:
http://t.co/THHo3lRM

OUTPUT:
HTTP/1.1 301 Moved Permanently
cache-control: private,max-age=300
date: Sat, 22 Jun 2013 11:16:28 GMT
expires: Sat, 22 Jun 2013 11:21:28 GMT
location: http://fb.me/1pNxOnFcT
server: tfe
Content-Length: 0

HTTP/1.1 301 Moved Permanently Location: http://www.huffingtonpost.com/2012/01/14/costa-
concordia-disaster-_n_1206167.html Content-Type: text/html; charset=utf-8
X-FB-Debug: rVCZC2NA//jyoJbrKkd7cPzOppS/2OyueM0IvLpBwts=
Date: Sat, 22 Jun 2013 11:16:28 GMT
Connection: keep-alive
Content-Length: 0

HTTP/1.1 200 OK
Server: Apache
P3P: CP='NO P3P'
Content-Type: text/html; charset=utf-8
Date: Sat, 22 Jun 2013 11:16:28 GMT
Connection: keep-alive
```