



UvA-DARE (Digital Academic Repository)

Regramming the Platform: Infrastructural Relations between Apps and Social Media

Gerlitz, C.; Helmond, A.; van der Vlist, F.N.; Weltevrede, E.

Publication date

2019

Document Version

Final published version

Published in

Computational Culture

License

CC BY-NC

[Link to publication](#)

Citation for published version (APA):

Gerlitz, C., Helmond, A., van der Vlist, F. N., & Weltevrede, E. (2019). Regramming the Platform: Infrastructural Relations between Apps and Social Media. *Computational Culture*, 7. <http://computationalculture.net/regramming-the-platform/>

General rights

It is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), other than for strictly personal, individual use, unless the work is under an open content license (like Creative Commons).

Disclaimer/Complaints regulations

If you believe that digital publication of certain material infringes any of your rights or (privacy) interests, please let the Library know, stating your reasons. In case of a legitimate complaint, the Library will make the material inaccessible and/or remove it from the website. Please Ask the Library: <https://uba.uva.nl/en/contact>, or a letter to: Library of the University of Amsterdam, Secretariat, Singel 425, 1012 WP Amsterdam, The Netherlands. You will be contacted as soon as possible.

Regramming the Platform: Infrastructural Relations between Apps and Social Media

ARTICLE INFORMATION

- **Author(s):** Carolin Gerlitz, Anne Helmond, Fernando N. van der Vlist, and Esther Weltevrede (alphabetical)
 - **Affiliation(s):** University of Siegen; University of Amsterdam
 - **Publication Date:** 21st October 2019
 - **Issue:** 7
 - **Citation:** Carolin Gerlitz, Anne Helmond, Fernando N. van der Vlist, and Esther Weltevrede (alphabetical). "Regramming the Platform: Infrastructural Relations between Apps and Social Media." *Computational Culture* 7 (21st October 2019). <http://computationalculture.net/regramming-the-platform/>.
-

ABSTRACT

In this article, we empirically analyse the infrastructural relations between mobile apps and social media platforms and present a methodology to account for app–platform relations. Contrary to previous research on platforms and apps, we develop our approach from the perspective of apps based on a relational understanding of infrastructure. Our app-centric approach to platforms and infrastructure provides critical insights into (i) the kinds of third-party apps developed on the peripheries of social media platforms, (ii) the diverse practices and features supported and extended by those apps, and (iii) the messy and contingent nature of the relations between apps and social media platforms. Our approach provides insights into alternative forms of platform programmability beyond APIs and into social media-based ‘innovation’ app ecosystems driven by creative developer workarounds. Drawing on quantitative and qualitative forms of analysis of Android and iOS apps related to Facebook, Instagram, Snapchat, and Twitter, we explore how third-party apps engage with the specific ‘grammars of action’ of social media platforms and outline five distinct forms of *regramming*. With regramming, we refer to how app developers work with and around the affordances, action grammars, and constraints imposed by platforms for using their data and functionality. We conclude with conceptual and methodological reflections on the infrastructural relations between apps and social media platforms, app stores, and mobile platforms from the perspective of apps.

Introduction

In May 2018, Instagram announced a new feature for users to share feed posts to their own Instagram Stories.¹ This practice of 'resharing', as Instagram calls it, had already been introduced over six years prior in a third-party app called *Regram*,² a popular app with over 500,000 downloads addressing the platform's lack of support for reposting images, or 'regramming'.³ "Why Regram?"] As a workaround to this limitation, *Regram* loaded a user's feed and added a Regram button underneath each post to instantly reshare and credit other people's posts. Since 2012, many apps have been developed for reposting Instagram content. Instagram finally introduced the popular practice to its own app, albeit in a limited form, as users can repost only to their Stories and not to their own feeds.⁴ Similarly, third-party Twitter clients introduced new platform features such as retweet and mute options that were later implemented into Twitter's official website and app.⁵ These examples suggest that there are continuous negotiations between developers of third-party apps and social media platforms, which sometimes adopt and sometimes defeat the added functionality. Moreover, these examples represent only a few of the platform-related apps currently available through app stores and external app directories.⁶]

In this article, we develop an empirical account of how apps build on platforms, focusing on Android and iOS mobile apps associated with four popular social media platforms: Facebook, Instagram, Snapchat, and Twitter.⁷ We propose an app-centric methodological approach for infrastructure analysis and argue that app stores are relevant entry points for research. Our twofold approach moves beyond previous studies that mainly focus on application programming interface (API) connections between apps and platforms, as we first explore how apps draw on platform practices and associated features to investigate the heterogeneous ways these connections are realised technically. The aim is to qualify the relations between apps and platforms beyond more general accounts of programmability of platforms or innovation ecosystems and to attend to the often messy, heterogeneous, mutual or conflictual relations that can emerge between apps and platforms. Drawing on Agre's work on capture through 'grammars of action',⁸ we advance the notion of *regramming* which focuses on the different ways in which app developers engage with platform data and functionality. We propose five forms of regramming the social media platform – intensification, reduction, revival, instruction, and transformation – which have implications for working, valorising, and interpreting app and platform data.

Platform Studies Meet App Studies

Studies of platforms, apps, and their relations largely appear in two fields of research: media studies and business and management studies. Within media studies, particularly in software and platform studies, there has been a growing interest in the materiality and technical underpinnings of social media as software platforms that mediate the interactions among multiple stakeholder groups with diverging interests and that enable third parties to use their data and functionality to build new services and applications.⁹ These approaches all seek to develop material accounts of software to contribute to an understanding of contemporary computational culture.¹⁰ Platform–app relations have been studied regarding a platform’s capacity to regulate and govern conditions for app development by focusing on controlled data exchanges through standardised platform functionality such as APIs and social plugins.¹¹ More recently, there has been increased dialogue between platform and infrastructure studies that explore the extent to which platforms are becoming infrastructural while infrastructures – robust, widely shared, accessible, and essential systems – are increasingly operating according to platform logics.¹² Meanwhile, apps are also considered to operate as platforms for developers to build on, as demonstrated by Facebook’s popular Messenger app and development platform.¹³ Central to these infrastructural perspectives are platform APIs,¹⁴ which provide a stable reference and common interface to developers for accessing platform data and functionality through standardised methods and formats. For developers, APIs pose an essential development infrastructure for building software on top of continuously changing platforms. For platforms, API connections render the process of building on top of platforms traceable and controllable, as they enable monitoring and regulating who makes connections and which data, functionality and practices are being built upon or altered. As Andreessen contends, APIs realise the programmability of platforms and their adaptability to custom needs, which may lead to new and unforeseen platform interpretations:

A ‘platform’ is a system that can be programmed 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.¹⁵

In their developer documentation, Facebook and Instagram invite third-party developers to use their platform resources to ‘Add something unique to the

community'¹⁶ while ruling out the replication of their core functionality. Twitter similarly demands that developers 'Avoid Replicating the Core Twitter Experience',¹⁷ and Snapchat only recently opened a limited set of public APIs.¹⁸ While platform APIs impose de facto infrastructural standards through the data forms and functions they make available, they also allow for certain degrees of openness of interpretation. This previously called 'interpretative flexibility' concerns perceptions of what platforms, data, and features can be used for,¹⁹ raising the question of whether and how third-party apps offer different interpretations of data and functionality and whether they support alternative practices.²⁰

Within the fields of management and information systems, the interest in platforms concerns the socio-technical relations among core or 'keystone' platforms, connected apps, third-party app developers, and businesses, which are underpinned by technological infrastructures.²¹ Similar to media studies approaches, management and information systems researchers adopt a platform-centric perspective with empirical contributions to with empirical contributions to the understanding of leading software platforms – including industry platforms, social media, app development platforms, and app stores – and their platform-based 'innovation ecosystems', which consist of developers, enterprise business firms, and apps connected to platforms through APIs.²² These apps are often built on top of other platforms as external contributions but may also be embedded within them. Here, platforms are understood as extensible codebases that foster 'generativity' by stimulating external contributions and innovation through hosting open architectures and platform resources that document how to develop applications for them.²³ Most platforms offer various technical resources, such as APIs and software development kits (SDKs), as well as non-technical resources, such as developer documentation and platform policies, to enable third-party app development while simultaneously constituting their means of control.²⁴ An app, in this specific context, is an 'add-on software subsystem or software service that connects to a platform to extend its functionality,' and it figures as a complementary good or service presumed to add value to that platform or its 'ecosystem'.²⁵ However, not all apps contribute equally to the platform, and some may not even align with its objectives, raising the question of how exactly those apps relate to or complement platforms.

The two approaches share platform-centric perspectives that yield insights into the role of the core platform in facilitating and governing the relations between platforms – which serve as the underpinning technological infrastructures – and

connected third-party apps. In contrast, we examine app–platform relations from the perspective of apps and consider what apps do with platforms’ functionality and data. This approach is becoming increasingly relevant because large numbers of apps are not created as standalone software applications but are built on top of, or in relation to, social media platforms, which raises critical questions around power relations (technical, economic, legal).²⁶ Many apps, for example, establish connections with remote hosts or servers to request content, serve ads, and connect to social media platform APIs to retrieve structured social data or embed social plugins. Starting from apps enables an empirical, open-ended exploration of such infrastructural relations and allows us to ask what apps do to platforms and vice versa.

Typically, platform–app relations are studied through API-based connections, for example, by scoping apps built on top of platforms listed in public APIs and app directories, such as ProgrammableWeb and IFTTT.²⁷ However, this approach presumes infrastructural relations that are API based and leaves out multiple non-formalised ways of relating to platforms – which we seek to identify in this article. Therefore, we start by delineating apps related to social media platforms using the leading Android and iOS app stores: Google Play and Apple’s App Store. Our analytical approach employs Star and Ruhleder’s relational understanding of infrastructure, asking ‘*when*—not *what*—is an infrastructure’.²⁸ In their view, infrastructure is not just a layering of technical standards but ‘shapes and is shaped by the conventions of a community of practice’.²⁹ Drawing on Agre, we suggest that these conventions are formalised into ‘grammars of action’ in the forms of posts, shares, likes, and followings.³⁰ At the same time, action grammars determine the action possibilities for users while simultaneously capturing data about these actions.³¹ The action grammars of platforms are standardised in form, transforming user content and interactions into data that can be compounded, commensurated, and combined within platforms’ databases.³² However, the practices and features supported by these action grammars are not standardised and come with a certain degree of ‘interpretative flexibility’ in meaning, value, and use scenarios for other user groups, such as app developers.³³ The aforementioned *Regram* app not only supports a widespread image-reposting practice on Instagram but also raises the question of how apps intensify, alter or regram platforms’ action grammars to accommodate the alternative ‘needs and niches’ of their user communities.³⁴

App-Centric Methodological Approach

We turn to app stores as the primary and 'native' environments for the distribution and monetisation of mobile apps in order to identify apps related to and relevant to platforms. Serving both as markets and indices of apps, app stores can be used to scope collections of apps in various ways.³⁵ We focus on Google Play and Apple's App Store, which are the two most popular app stores, and query them both for apps associated with four leading social media platforms: Facebook, Instagram, Snapchat, and Twitter. We focus on Facebook and Twitter, as they have been around for more than ten years and have mature platforms for developers and businesses. Moreover, Facebook and Twitter have received considerable scholarly attention in platform-centric studies, and we aim to supplement these studies from an app-centric perspective. To complement our comparison, we include Instagram and Snapchat, as they were originally launched as mobile apps and only later grew into platforms for developers, businesses, and advertisers.³⁶ However, while we explore app-platform relations in two dominant app stores, the aim is not to perform a comparative analysis of stores but rather to use the inventories of both stores to scope a wide range of Android and iOS apps, building on previous research that deployed the capabilities of app stores for demarcating source sets.³⁷ Moreover, this gives us more insights into the different forms of management and regulation that app stores perform, which influence the kinds of relations that are allowed to emerge between platforms and apps in the first place. Google Play and the App Store have specific logics and mechanisms for organising, sorting, ranking, relating, and recommending apps. In addition, they each provide different forms of data access that we needed to account for. Additionally, Google Play has a web interface that can be scraped, while the App Store has a limited web interface but can be accessed programmatically via the iTunes Search API.

First, we used the search capabilities of Google Play and the App Store by querying them for [Facebook], [Instagram], [Snapchat], and [Twitter]. To prevent personalisation and localisation in the results returned, we used internet proxies to default the locale to the United States and to English language settings. In total, Google Play yielded 998 apps across all four search queries, and the App Store yielded 531 apps (Table 1). For each app listing, app stores provide a number of details, including app titles, descriptions, and various technological specifications. Most of these details are written by developers themselves, and there are app store guidelines that stress the importance of accurate and focused titles and descriptions to cover what the app is about.³⁸ This is important information for our analysis because we use these titles and descriptions for identifying the key practices, features, and

platform connections of apps. Second, to enlarge these source sets with additional relevant apps that were not returned by the initial search queries, we subsequently followed each app’s recommendations. Notably, app stores employ different kinds of algorithmic and personalised recommendations to recommend related or similar apps to users.³⁹ Google Play shows ‘Similar Apps’ based on topics derived from words and phrases in the titles and descriptions of apps – creating relations based on topical clusters. The App Store, however, lists related apps under ‘You May Also Like’ (specified in the URL as ‘customers-also-bought’), which are based on other apps purchased, downloaded, or installed by users – creating relations based on use practices. Taking all related apps listed for our initial search results, the Google Play source set led to a total number of 12,772 unique apps and the App Store to 5,180 (Table 1).⁴⁰ We collected the names and details of all these apps (e.g., bundle identifiers, app store categories, developer names, descriptions, ratings, reviews, prices, software versions) using programmatic data retrieval tools and custom-built web scrapers.⁴¹

Table 1. Unique apps per source set.

Search query	Google Play		App Store	
	Search	‘You May Also Like’	Search	‘You May Also Like’
[Facebook]	250	2,419	161	1,736
[Instagram]	250	2,906	106	772
[Snapchat]	250	3,673	130	1,369
[Twitter]	248	3,774	134	1,303

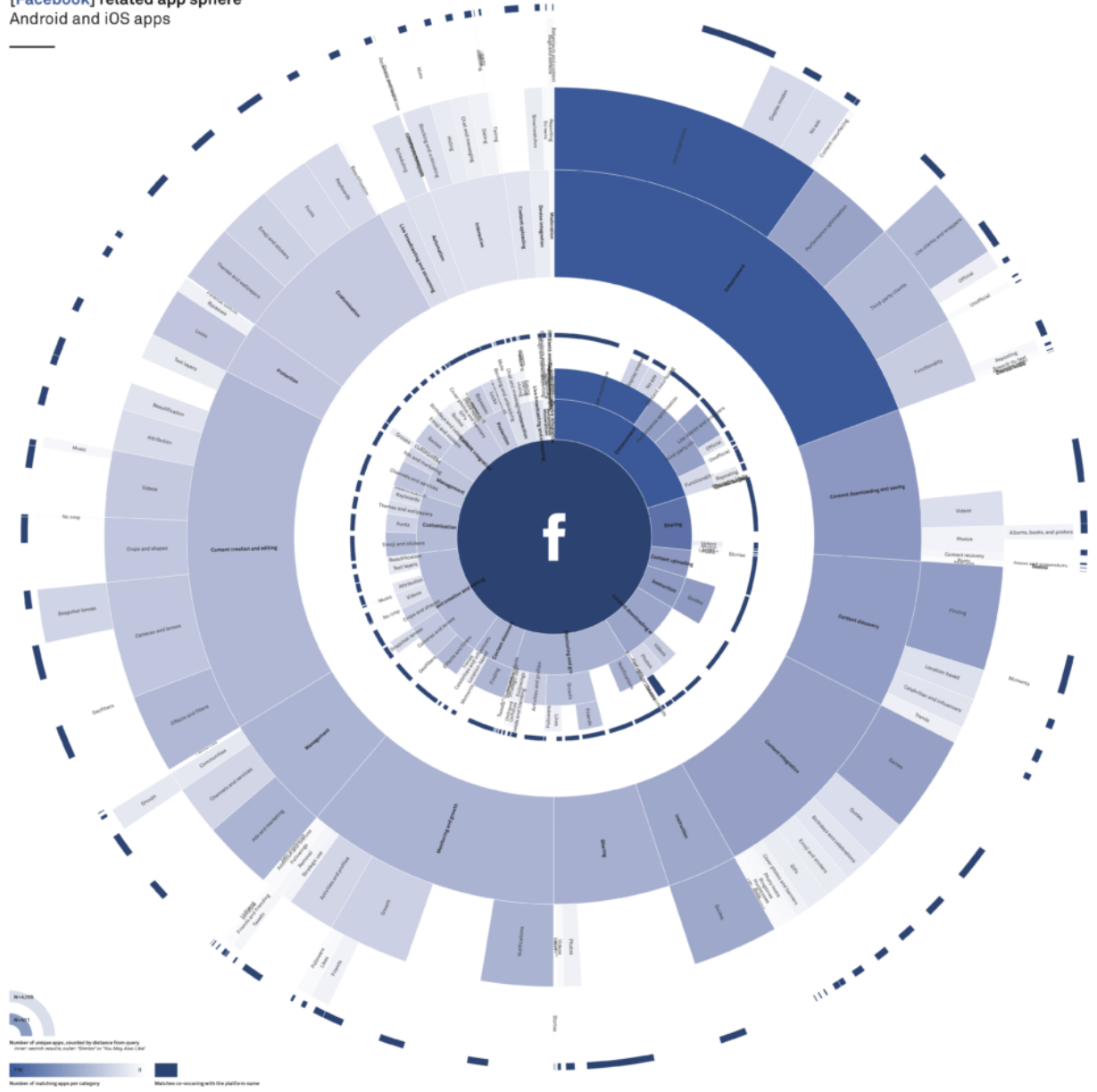
To analyse how apps engage with platforms’ action grammars, we developed a scalable methodological approach that involves a close reading of app titles and descriptions followed by an emergent, collaborative coding process. The resulting lexicon subsequently informed a computational analysis and classification process. Much like the computational topic classifier models used for identifying Google Play topic clusters, we identified prominent words and phrases describing our apps and subsequently determine broader topics to cluster and interpret them. For our purposes, these broader topics are of three different kinds: platform-related practices (e.g., monitoring and growth, content creation), general features across platforms (e.g., growth, effects and filters), and platform-specific features (e.g., friends, followers). This twofold focus on practices and features allows us to determine the forms of functionality present in apps and how they relate to platform-specific action grammars. To detect the resonance of each identified practice and feature in our entire source set, we

created search patterns with extended regular expressions that enabled us to run advanced pattern-based searches across all four source sets.⁴² In a second step, we also ran nested searches to further contextualise each positive match when it co-occurred with one of the social media platform names (i.e., Facebook, Instagram, Snapchat, Twitter), which helped us distinguish more and less relevant matches. The outcome is a frequency count for each classified practice and feature (per store, per source set), which enables us to further qualify the relation apps set up to the associated platforms.

Classification of App–Platform Relations

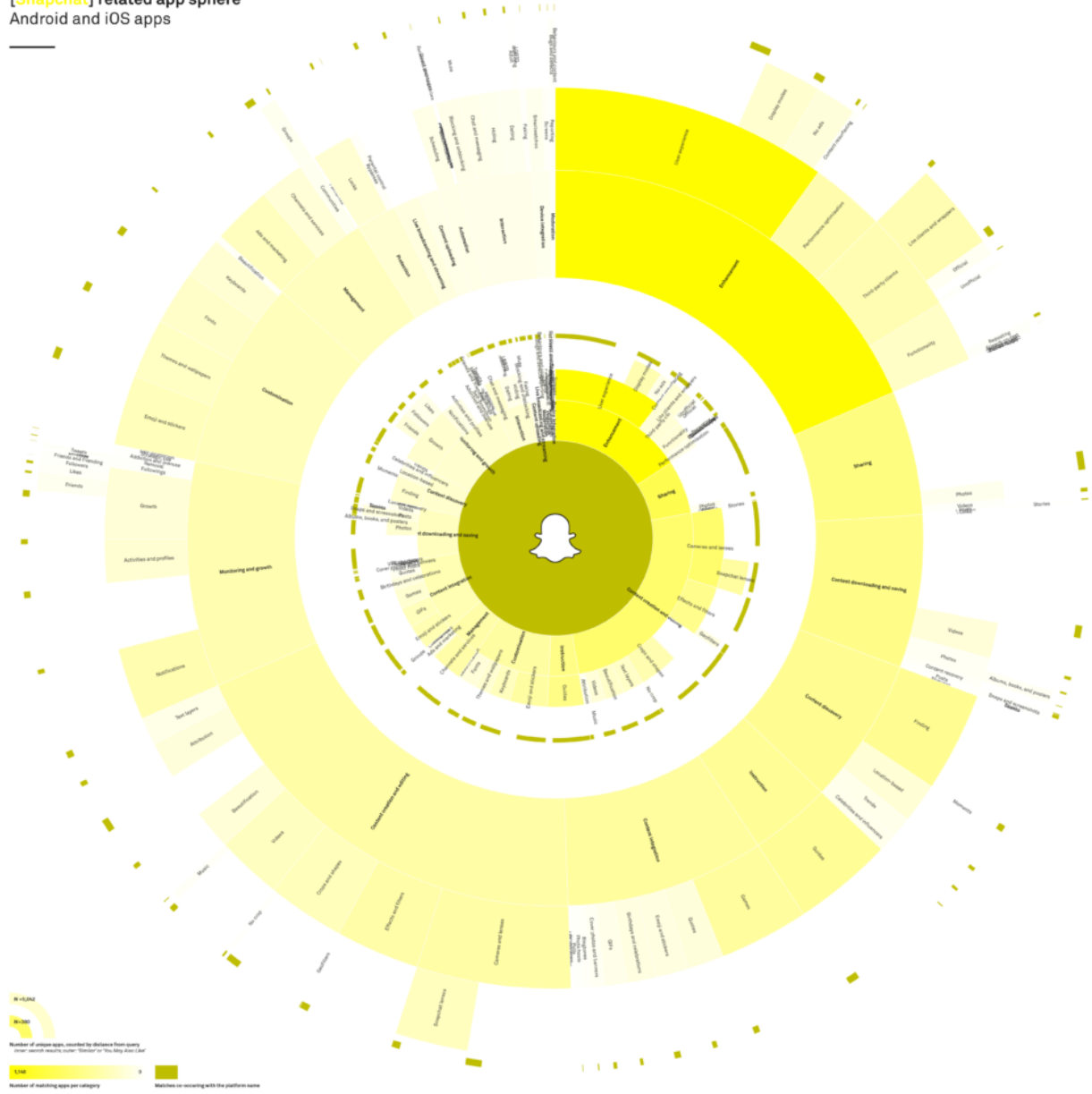
As we have found, apps engage with platforms in multiple ways, which we captured in the following visualisations. Figures 1a–1d provide aggregate views of the practices that these apps offer and how they are distributed per source set. Each figure combines two hierarchical sunburst diagrams, displaying the various associated practices and features per social media platform. The inner sunbursts show the results for the smaller source set with initial search results only [Facebook], [Instagram], [Snapchat] and [Twitter], and the outer sunbursts show the results for the much larger – and inevitably less relevant – source sets of algorithmically related apps ('Similar', 'You May Also Like'). The rings differentiate between the platform-related practices that apps engage with (the innermost rings) as well as the platform features supporting these practices (the outer rings). The more each practice or feature resonates with the source set of apps, the larger its relative size. Finally, the thinnest rings detail how many apps co-occur with the initial search query for that source set – that is, with the social media platform names – indicating stronger app–platform relations. Overall, apps sourced directly from the initial search queries have much higher relevance, whereas algorithmically related apps score lower due to topic drift. Instagram-related apps score the highest on relevance overall (avg. 87.86%; Figure 1b).

[Facebook] related app sphere
Android and iOS apps



[Click to enlarge.]

[Snapchat] related app sphere
Android and iOS apps



[Click to enlarge.]

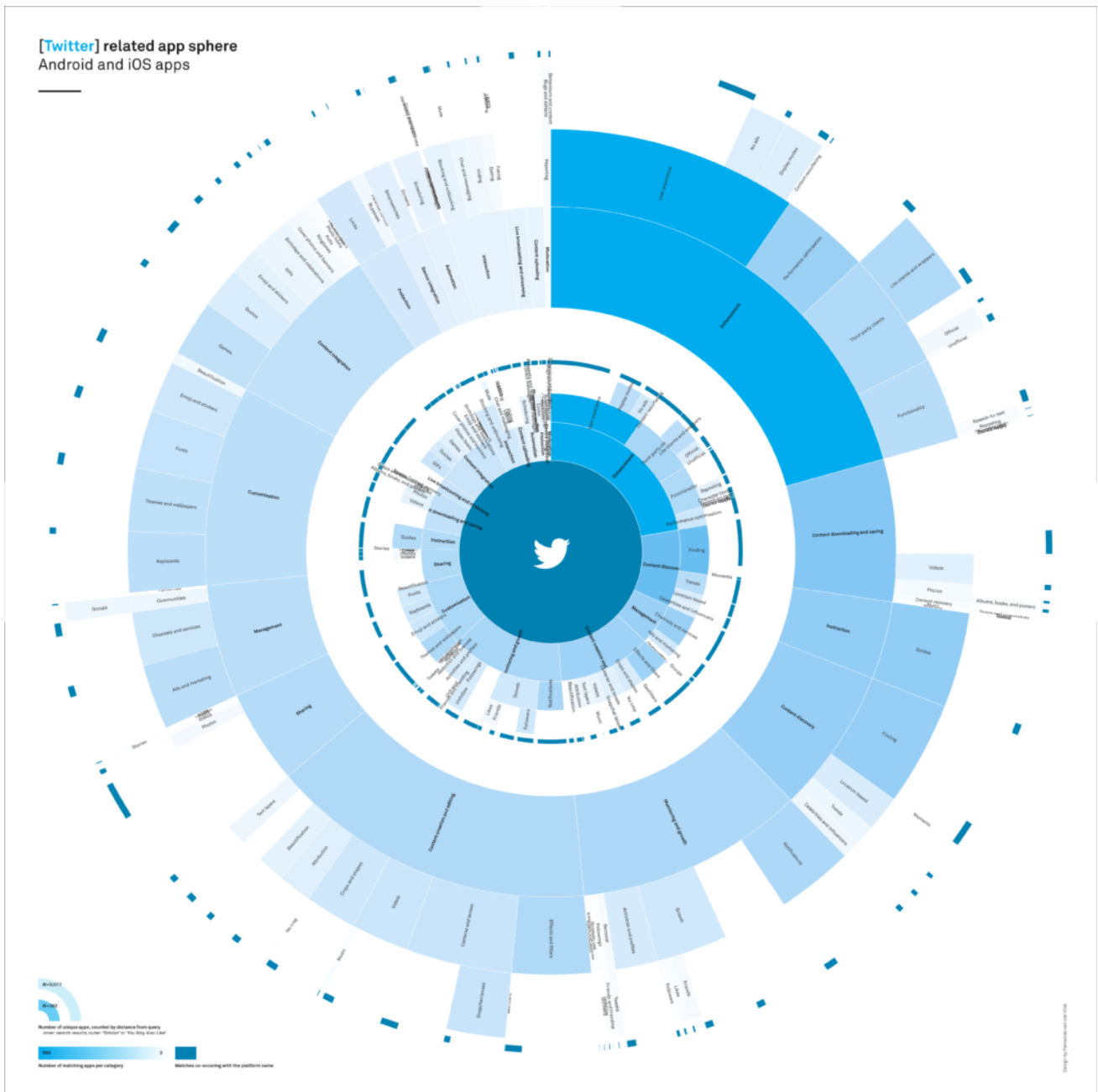


Figure 1a–1d. Sunburst diagram of the [Facebook] ($N = 4,566$), [Instagram] ($N = 4,034$), [Snapchat] ($N = 5,422$), and [Twitter] ($N = 5,459$) related app spheres for both Android and iOS apps. Inner rings display categorised search results for the initial search queries and the outer rings display their algorithmically-related apps. Hierarchy: practices, first-level features, and second-level features; size and colour-coding: by frequency count of matching apps (linear scale). [Click to enlarge.]

Platform-related practices and features

This form of quantitative and qualitative analysis offers an aggregated perspective on the platform-related practices these apps offer to their users (Figure 2a). As such, it complements more common, small-scale studies of single apps and their specific use practices, interfaces, and users.⁴³ Analysed apps relate to action grammars in multiple ways. Most prominently, the majority of apps do not rework but enhance the existing platform functionality and user experience or complement established content creation practices

across all four source sets. Many apps accommodate analytics practices such as social monitoring and growth or management functionality to drive popularity and presence – supporting practices that are in line with platform objectives. A smaller proportion of apps enhance platform functionality that supports practices such as content discovery, sharing, uploading, (live) streaming, and downloading and saving content. Further, there are apps that focus on providing instructional guides, protection, and screen locks.

In addition to practices that are shared across multiple platforms, the visualisations provide insights into platform-specific practices and features. For example, such specificity is found in the practice of integrating connected devices in Twitter-related apps, content creation and editing in Snapchat-related apps, and content integration and discovery in Snapchat and Twitter-related apps, while content uploading, protection and screen locks resonate more with Facebook-related apps. Compared to the others, Instagram-related apps are the least distinctive in terms of their associated practices. Overall, though, Instagram and Snapchat are both focused more on original content creation and editing, and Twitter leads in terms of the number of alternative third-party clients.

On the feature level, however, the distinctions between platforms are more clearly articulated because features typically relate to specific functionality and action grammars (Figure 2b). There are clear differences in the content level of these platforms (e.g., visual, textual, video). Snapchat and Instagram are particularly distinctive in terms of content creation and editing apps. Snapchat-related apps include cameras, lenses, effects, and filters, while Instagram-related apps include 'no crop' apps and apps for adding text layers to images. 'No crop' apps address a distinctive aspect of Instagram, namely, that the official app automatically square-cropped images before it started allowing portrait and landscape formats.⁴⁴ In contrast, Facebook-related apps include birthdays and celebrations, games, and horoscopes, and Twitter has apps related to photo hosting, polls, quotes, and even ringtones.

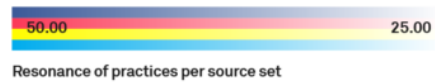
Additionally, some practices are shared but are constituted differently due to platform specificities (e.g., specific constraints, features, metrics). Apps for content discovery, for example, are articulated in platform-specific ways, including apps related to Twitter Moments, Trends, location-based discovery, and the discovery of celebrities or influencers on Facebook. Functionality enhancement resonates strongly for Facebook-, Instagram-, and Twitter-related apps. For Facebook and Instagram, such enhancement apps enable users to

replay Stories, share video Stories longer than fifteen seconds, or add music, while Twitter-related enhancement apps enable users to write posts beyond the character limit, post 'tweetstorms' (i.e., a series of related tweets posted successively), or dictate tweets. Furthermore, apps enhancing the user experience (e.g., resurfacing content, ad blocking) occur more frequently in the Snapchat source set, while Facebook and Twitter have many related apps for optimising device performance and reducing network traffic load. Such apps allow users to engage with platforms despite infrastructural connectivity constraints – possibly expanding the user bases of these platforms. Social monitoring and growth apps are featured prominently for all platforms with Twitter-related apps for unfollowing, content removal, and follower growth; Instagram-related apps for content removal, self-promotion, growing followers or likes, and monitoring overuse; Snapchat-related apps for monitoring one's following and tracking overuse; and Facebook-related apps for monitoring friend activity, profiles, unfriending, and apps assisting with strategic use (e.g., timing posts). Interestingly, Instagram features the most anti-addiction apps. Apps catering to businesses, advertisers, and marketers, including multi-account and multi-channel integration, occur most notably in relation to Twitter and Facebook. Finally, although sharing is a core aspect of social media in general,⁴⁵ there are differences in terms of the objects shared. Twitter-related apps share links, locations, and posts, while Facebook and Instagram-related apps share Stories, videos, photos, and links. Similarly, for content downloading or saving, the objects are different. While Facebook and Instagram-related apps have the most downloaders for photos, videos, Stories, and statuses, Snapchat has related apps for recovering content, saving Snaps, and downloading videos.

App–platform relations

Practices

Practices	Counts (% of row)				N
	[Facebook]	[Instagram]	[Snapchat]	[Twitter]	
Automation	23.36	25.00	21.97	29.67	792
Content creation and editing	22.39	23.24	30.02	24.35	8,004
Content discovery	23.83	19.13	27.31	29.72	4,108
Content downloading and saving	23.92	22.27	27.51	26.30	4,301
Content integration	26.49	17.39	29.88	26.23	5,043
Content uploading	29.64	25.78	22.67	21.92	803
Customisation	21.93	20.93	28.14	29.00	4,310
Device integration	21.11	18.49	24.81	35.59	649
Enhancement	23.73	21.44	26.56	28.27	9,969
Instruction	22.56	19.26	29.63	28.55	2,487
Interaction	26.39	17.80	29.68	26.14	1,978
Live broadcasting and streaming	26.84	20.94	26.96	25.27	831
Management	25.35	19.37	25.60	29.67	3,242
Moderation	20.20	23.74	23.74	32.32	198
Monitoring and growth	25.11	20.67	26.30	27.92	5,921
Protection	29.21	20.83	23.96	26.00	1,373
Sharing	23.66	24.67	29.07	22.60	3,681



Design by Fernando van der Vliet

[Click to enlarge.]

App-platform relations

Practices and features

Practices	Features		Counts (% of row)				N
	First-order	Second-order	[Facebook]	[Instagram]	[Snapchat]	[Twitter]	
Automation	Answers and replies		29.41	23.53	17.05	29.41	17
Automation	Beautification		11.54	26.92	42.31	19.23	26
Automation	Feeds and timelines		10.00	30.00	30.00	30.00	10
Automation	Posts and messages	Direct messages	0.00	0.00	0.00	100.00	7
Automation	Posts and messages		27.27	27.27	18.18	27.27	11
Automation	Posts and messages	Retweets and favourites	21.74	0.00	26.09	52.17	23
Automation	Scheduling		24.06	25.73	21.14	29.07	719
Content creation and editing	Attribution		24.64	26.09	25.51	23.76	1,031
Content creation and editing	Beautification		20.42	17.73	36.43	25.43	818
Content creation and editing	Cameras and lenses		20.76	27.26	32.72	19.26	2,124
Content creation and editing	Cameras and lenses	Snapchat lenses	20.67	18.20	34.57	26.56	1,374
Content creation and editing	Crops and shapes		22.57	23.82	31.70	22.11	1,719
Content creation and editing	Crops and shapes	No crop	14.41	39.83	31.36	14.41	118
Content creation and editing	Effects and filters	Geofilters	9.38	18.75	59.38	12.50	32
Content creation and editing	Effects and filters		22.64	21.42	28.54	27.40	2,204
Content creation and editing	Text layers		16.22	32.06	34.71	17.01	1,017
Content creation and editing	Text layers		20.66	31.73	27.06	20.54	813
Content creation and editing	Videos	Music	28.22	29.21	23.76	18.81	202
Content creation and editing	Videos		23.19	23.12	28.11	25.58	1,905
Content discovery	Celebrities and influencers		32.91	17.09	23.20	26.80	556
Content discovery	Finding	Moments	14.29	14.29	14.29	57.14	7
Content discovery	Finding		24.09	18.97	26.89	30.25	2,694
Content discovery	Location-based		20.02	20.13	30.88	28.97	939
Content discovery	Trends		19.27	17.73	29.09	33.91	519
Content downloading and saving	Content recovery		25.82	24.73	27.47	21.98	182
Content downloading and saving	Content recovery		23.82	19.86	27.54	28.77	2,930
Content downloading and saving	Photos	Albums, books, and posters	29.93	24.34	26.64	19.08	304
Content downloading and saving	Photos		23.11	30.19	24.76	21.93	424
Content downloading and saving	Posts	Snap and screenshots	14.78	18.26	52.17	14.78	115
Content downloading and saving	Posts	Stories	23.08	34.62	19.23	23.08	26
Content downloading and saving	Stories		28.57	26.53	16.33	28.57	49
Content downloading and saving	Videos		23.16	27.95	26.02	23.47	980
Content integration	Birthdays and celebrations		28.68	17.90	31.44	21.98	687
Content integration	Cover photos and banners		23.44	25.52	24.22	26.82	384
Content integration	Emoji and stickers		20.50	19.88	32.75	26.88	800
Content integration	Games		31.62	11.99	33.95	22.44	1,885
Content integration	GiFs		21.68	23.05	27.94	27.33	695
Content integration	Horoscopes		53.23	4.84	25.81	16.13	62
Content integration	Photo hosts		23.14	28.38	18.34	30.13	229
Content integration	Polls		21.09	22.66	24.22	32.03	128
Content integration	Quotes		25.00	18.20	25.70	31.10	1,000
Content integration	Ringtones		21.64	10.53	30.41	37.43	171
Content integration	URL shorteners		26.32	26.32	21.05	26.32	19
Content uploading			29.64	25.78	22.67	21.92	803
Customisation	Beautification		13.14	18.86	25.14	42.86	175
Customisation	Emoji and stickers		21.09	22.89	32.00	24.03	1,869
Customisation	Fonts		21.18	23.11	26.59	29.12	1,487
Customisation	Keyboards		19.68	14.78	26.77	38.77	1,184
Customisation	Themes and wallpapers		22.47	19.57	27.27	30.68	1,584
Device integration	Screens		24.29	25.71	32.14	17.85	140
Device integration	Smartwatches		20.23	17.96	22.68	39.13	529
Enhancement	Functionality	Character limits	34.21	7.89	13.18	44.74	38
Enhancement	Functionality		20.92	17.42	25.72	35.94	1,458
Enhancement	Functionality	Posting	20.94	45.94	13.13	20.00	320
Enhancement	Functionality	Stories length	28.57	28.57	14.29	28.57	7
Enhancement	Functionality	Stories replay	42.86	42.86	14.29	0.00	7
Enhancement	Functionality	Speech-to-text	22.32	10.71	24.11	42.86	112
Enhancement	Functionality	Tweetstorms	0.00	14.29	14.29	71.43	7
Enhancement	Performance optimisation		25.88	20.87	23.73	29.51	2,233
Enhancement	Third-party clients	Lite clients and wrappers	23.25	21.24	27.04	28.47	2,034
Enhancement	Third-party clients		23.76	12.43	15.75	48.07	362
Enhancement	Third-party clients	Official	25.87	17.72	21.45	34.97	429
Enhancement	Third-party clients	Unofficial	27.78	15.87	30.16	26.19	126
Enhancement	User experience	Content resurfacing	23.90	28.93	27.04	20.13	159
Enhancement	User experience	Display modes	20.81	28.21	28.81	22.17	1,326
Enhancement	User experience		24.20	21.38	27.53	26.90	4,969
Enhancement	User experience	No ads	24.40	19.28	28.10	28.21	918
Instruction	Guides		22.56	19.26	29.63	28.55	2,487
Interaction	Blocking and unblocking	Mute	23.08	12.82	10.26	53.85	39
Interaction	Blocking and unblocking		26.89	17.52	23.87	31.72	662
Interaction	Chat and messaging		25.31	14.02	34.73	25.94	478
Interaction	Dating	Adult	20.00	11.11	53.33	15.56	45
Interaction	Dating	LGBTQ	40.00	0.00	60.00	0.00	5
Interaction	Dating	Matching	25.00	4.55	50.00	20.45	44
Interaction	Dating		27.90	17.65	33.43	21.73	359
Interaction	Faking		25.44	21.05	38.60	14.91	114
Interaction	Hiding		26.39	22.96	28.11	22.53	466
Live broadcasting and streaming			26.84	20.94	26.96	25.27	831
Management	Ads and marketing		27.01	16.00	26.57	30.42	1,581
Management	Channels and services		22.95	22.46	24.48	30.11	1,438
Management	Communities	Groups	28.37	20.23	24.42	26.98	430
Management	Communities		26.44	26.44	13.79	33.33	87
Moderation	Reporting	Behaviours and content	23.08	15.38	46.15	15.38	13
Moderation	Reporting	Bugs and defects	20.53	23.68	22.63	33.16	190
Monitoring and growth	Activities and profiles		26.09	19.95	26.52	27.44	1,188
Monitoring and growth	Addiction and overuse		24.00	14.00	38.00	24.00	100
Monitoring and growth	Followings	Friends and friending	30.99	14.08	30.99	23.94	142
Monitoring and growth	Followings	Unfollow	20.31	28.13	11.72	39.84	128
Monitoring and growth	Followings	Unfriend	33.33	29.63	0.00	37.04	27
Monitoring and growth	Growth	Followers	17.79	31.62	20.55	30.94	253

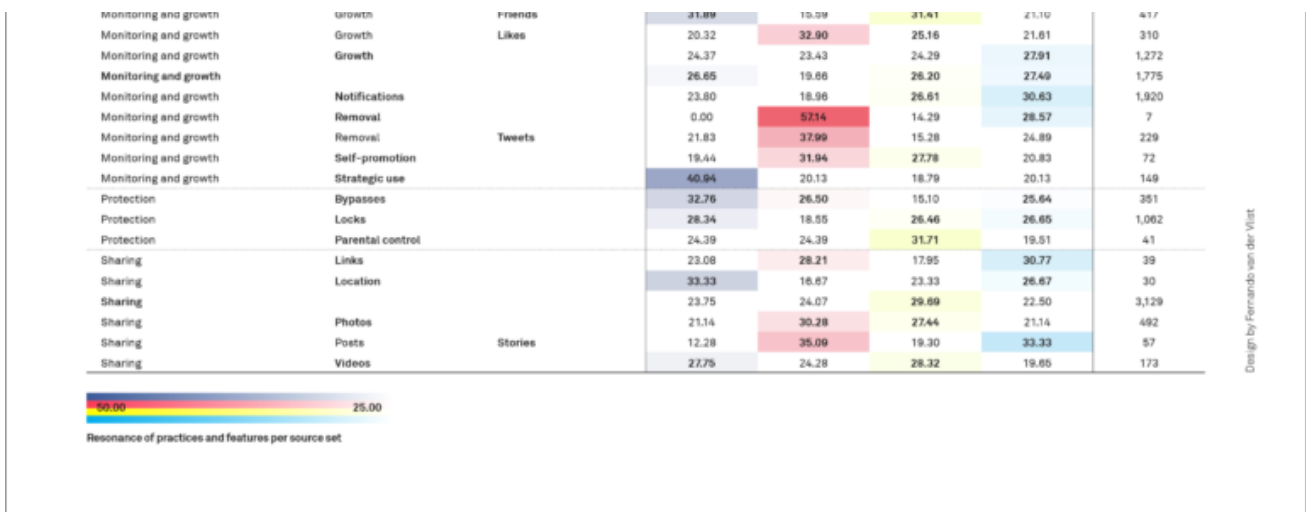


Figure 2a–2b. App–platform relations, grouped by practices. Each row is a percentage corresponding to the count of matching apps per source set and per row. Values higher than 25.00% are indicative of specificity and the higher the values, the more specific the corresponding practices or features are to each source set. Colour-coding: by frequency count of matching apps (linear scale). [Click to enlarge.]

Platform relations

The analysis of app titles and descriptions has shown the multiple ways in which apps can relate to, enhance, advance, alter or add to platform practices and their features. In the next step, we inquire how these relations are realised technically and infrastructurally – whether apps connect to platforms through API and other HTTP requests or other means and whether these connections are subject to platform regulations (Table 2). Interestingly, a considerable number of app developers have included legal disclaimers in their descriptions to distance themselves from the official platform – in part because they know their apps might violate platform policies. Only very few developers describe how the technical connections between their apps and platforms are realised. Some practices are technically supported through platform-specific action grammars that can be programmatically accessed through APIs and SDKs. However, not all of the social media platforms in this study provide such programmatic access to third parties. Facebook and Twitter both offer APIs and SDKs to third-party developers and business developers, while Instagram recently shut down its Platform APIs for third-party developers in favour of its new Graph API for business developers only.⁴⁶ Snapchat also favours business developers, as it offers resources such as Marketing APIs and Lens Studio to business partners only.⁴⁷

Table 2. Detected app–platform relations per source set.

Relation	[Facebook]	[Instagram]	[Snapchat]	[Twitter]
Brand (mentions)	1,449 (34.96%)	2,945 (80.03%)	614 (12.17%)	1,107 (21.80%)

Legal (mentions)	302 (7.29%)	318 (8.65%)	268 (5.31%)	305 (6.01%)
Technical (mentions)	61 (1.47%)	62 (1.68%)	70 (1.39%)	114 (2.24%)
Technical (libraries, SDKs)*	83 (33.20%)	0 (0%)	0 (0%)	40 (16.13%)
Technical (HTTP requests)*	156 (62.40%)	89 (35.60%)	12 (4.80%)	102 (41.13%)

* Only for Google Play search results ($N = 998$).

To explore how apps connect to platforms on an infrastructural level, we employ static analysis of Android package files (.apk) and examine the contents of the Android apps in our source set using their decompiled source code.⁴⁸ This allows us to detect official software libraries embedded within apps as well as remote network requests made to platforms, which are required to connect with social media platforms' APIs and to load, embed, and wrap platform content in third-party apps. In other words, when we detect traces of such connections, we know that apps have embedded official resources to integrate platform data and functionality, and conversely, if they do not, we know that there is no official technical or API-based integration (although unofficial resources may be used).

We find that 33.20% of all Facebook-related apps contain the official Facebook Android SDK,⁴⁹ and 16.13% of Twitter-related apps contain the Twitter Kit,⁵⁰ which are both software libraries for integrating platform content and functionality in external apps. As Snapchat and Instagram did not offer public SDKs at the time of data collection, no results were retrieved. Another story unfolds when we look at the HTTP requests made by our Android apps to the social media platforms. These apps make network requests to platforms' public APIs, private APIs, and mobile pages and request content from content delivery networks. Nearly two-thirds (62.40%) of Facebook-related apps make HTTP requests to facebook.com (e.g., graph.facebook.com, m.facebook.com), to sign in and authenticate with Facebook, and to retrieve data from Facebook's mobile site, on which the majority of lightweight clients are built. Over 41% of Twitter-related apps make requests to the platform's APIs and mobile site (e.g., api.twitter.com, mobile.twitter.com), and over 35% of Instagram-related apps connect to the platform's public or private APIs (e.g., api.instagram.com, i.instagram.com). Only 4.8% of Snapchat-related apps make requests to the platform; this is because there was no public API at the time of data collection, and Snapchat actively blocks third-party applications from accessing its private API.⁵¹ Direct connections are indicative of data exchanges between apps and platforms, but even without direct connections, platform-related apps can influence content creation, interpretation of platforms' action grammars, and

envisioned scenarios of use, as we detail in the next section.

We can further distinguish between official and unofficial APIs and SDKs. For example, although `api.instagram.com` is Instagram's official (i.e., public and documented) API, we find many apps making requests to `i.instagram.com`. The latter refers to the platform's unofficial (i.e., private, undocumented) API, which represents a developer workaround to overcome, bypass, or minimise limitations posed by official platform infrastructures, such as inaccessible data fields or rate limiting.⁵² These findings indicate that Facebook most tightly regulates app development on its platform and that Facebook-related apps have the most technical relations. There are also differences in terms of how platforms treat and regulate the existence of unofficial APIs and SDKs, with Snapchat explicitly forbidding unauthorised access to any of its services and Facebook prominently listing popular unofficial SDKs 'built by amazing communities of active developers' on their developer pages.⁵³

However, most connections that we find represent other kinds of infrastructural relations beyond these APIs and SDKs. This suggests that the programmability of social media platforms is not merely manifested through APIs but may also take other forms. In what follows, we further explore app–platform relations by enquiring how apps relate to platforms' action grammars.

Five forms of regramming

Our results suggest that most apps do not use platforms' official software libraries or connect to official APIs. Consequently, the kinds of app–platform relations that emerge are not always technical and challenge common ideas around API-based programmability. We understand app developers' engagements with social media platforms' action grammars as forms of regramming that are not necessarily aligned with the interests and objectives of those platforms. These forms reflect different ways for app developers to engage with platform functionality and data. We find five such forms of regramming: intensification, reduction, revival, instruction, and transformation. Apps accommodate diverse use practices and their developers can regram platforms in different ways. In this sense, regramming differentiates and qualifies existing notions of programmability, as the dimensions of regramming signpost what apps do to platforms and their data. However, because social media and app stores both have their own policies and terms of service, these developer contributions are not always welcomed and therefore reveal the messy, contingent peripheries of platforms.⁵⁴

Intensification

One group of apps offers functionality that seemingly aligns with platform interests and implements only minor adjustments that further intensify existing use practices. That is, the app's functionality can be distinct from the platform's functionality but is nonetheless captured within the existing logics and grammars of the core platform, thus further intensifying its native use practices. As Gehl has argued, social media platforms are designed as 'empty frames' that await user-generated content to add value to the platform.⁵⁵ Therefore, it is interesting that most apps in this category engage on the level of content creation and editing.

First, most intensification concerns visual content. Featuring prominently in all source sets, content creation and editing apps such as *Beautify*⁵⁶ and *Snapseed*⁵⁷ enable users to create, edit, and 'beautify' visual content before posting online. For Instagram and Snapchat, such apps address photo or video content editing, cameras and lenses, filters, face masks, croppings, and collages. They support means of content creation focused on specific aesthetic formats. Creative expression is structured and channelled through recurring visual grammars, such as popular filters, animal faces, thematic masks, and face swaps. The question of aesthetic standardisation is particularly apparent in the proliferation of apps for facial alteration features, skin smoothing, hairstyle alterations, and the application of thematic make-up. As such, these apps offer various pre-sets for photo, video, and selfie editing; before that content is posted online using one of the available posting methods. However, while visual content creation and editing are widely accommodated by third-party apps, there is notably less support for creating and editing non-visual content forms. This is remarkable since previous research on third-party Twitter apps using API data has revealed an ecosystem of web-based apps supporting textual content creation and editing (e.g., RSS feed integrations, news aggregation tools).⁵⁸

Second, a significantly smaller number of intensification apps focus on textual content forms. For platforms such as Facebook and Twitter, which also feature textual content forms, there are apps for the discovery of inspirational quotes, selection of hashtags for audience growth purposes, and customised emoji and stickers (implemented with device-level keyboards). Additionally, there are apps to support the automatic generation of posts, answers, and replies. Such apps intensify the use of action grammars by increasing the volume of content created, shared, and engaged with while simultaneously creating format standards and templates. They are not necessarily concerned with content

quality at all (although Snapchat has some 'auto-beautification' apps). Twitter leads in terms of the number of such automation apps, many of which are professional social media management apps such as *Hootsuite – Social Media Tools*⁵⁹ and *Everypost for Social Media*.⁶⁰ While it may seem that Twitter has the most automation-friendly use culture, the platform has become increasingly restrictive, as reflected in its 'Automation Rules.'⁶¹ Because these apps can operate against Twitter's interests, Twitter is continuously monitoring its APIs to detect policy violations.⁶² This recently led to API access suspensions for apps supporting bulk user following and unfollowing on Twitter, such as *Statusbrew*⁶³ and *Crowdfire*⁶⁴, which are featured in both our Android and iOS source sets.⁶⁵

Finally, the intensification of action grammars not only concerns content-related functionality but also targets the reputation mechanisms and performance metrics designed to optimise users' audience reach and strategic engagement. Twitter and Snapchat had the most monitoring and growth-related as well as management-related apps. These apps mostly aligned with platforms' own interests and enable professional users such as businesses and 'influencers' to 'understand and manage their audience', to 'develop their content strategy' in Instagram's case,⁶⁶ and to 'publish and analyze Tweets, optimise ads, and create unique customer experiences' in Twitter's case.⁶⁷ All source sets include business-oriented apps for monitoring and management-related needs, including ones that do not use official platform resources. In the cases of Facebook, Instagram, and Twitter there is a focus on monitoring users who do not follow, have unfriended, or have unfollowed the focal user. Such information is not officially offered to users of those platforms, although it is important for these users because they seek to engage strategically and grow their audiences or engagement. Then, there are apps that offer interfaces to connect and use multiple user accounts or channels at once or to enable ad management, team engagement, and community management. These capabilities feature prominently for Facebook, Twitter, and Snapchat and do not necessarily alter or introduce alternative action grammars; rather, they assist existing users in more effectively leveraging social media for strategic purposes. These strategic practices are often part of business practices that intensify existing use practices and therefore largely align with platforms' own interests in growing their multi-sided markets. That is, multi-sided social media platforms mediate the interactions between, on the one hand, billions of end-users sharing and engaging with content or followers and, on the other hand, millions of professionals seeking end-users' attention to their benefit. Here, too, platforms are actively monitoring the functionality of these third-party apps for growing audiences and engagement. For example, Instagram claims to remove

'inauthentic likes, follows and comments from accounts that use third-party apps to boost their popularity' when these apps violate community guidelines, policies, or terms of service.⁶⁸

Reduction

The second group of apps reduces existing functionality to improve of specific use practices. These apps make use of the existing functionality offered by platforms but implement them in reduced form or make minor adjustments to improve their situative performance in some way.

Most of these apps are alternative, unofficial third-party 'clients' for Facebook and Twitter offering lightweight alternatives to official platform clients to free up device storage, limit memory and data usage, and increase battery life. These apps negotiate the infrastructure dependencies and limits on the levels of the device and network, enable better performance in low-bandwidth regions and on low-end mobile devices, and reduce the cost of mobile data plans. In the case of Facebook, many of these apps are 'wrappers', which load content and functionality from Facebook's mobile, zero, and free basic sites within their own apps (m.facebook.com, mobile.facebook.com, 0.facebook.com, mbasic.facebook.com). These apps simply repackage existing – and already reduced or optimised – content and functionality and then embed them within their own alternative Facebook clients. In the case of Twitter, however, these unofficial clients mostly call upon Twitter's official APIs (e.g., api.twitter.com). While some Twitter clients offer reduced content and functionality, some apps, such as *Tweecha Lite*,⁶⁹ also introduce their own functionality, such as options to use colour-coded labels or mark tweets as read. Unofficial clients, however, provide alternative experiences of platforms' own content and functionality, which again seems to be against platform policies. For example, Facebook's policy reads, 'Respect the way Facebook looks and functions. Don't offer experiences that change it',⁷⁰ and Twitter's policy asks developers and users to 'Maintain the Integrity of Twitter's Products'.⁷¹ Despite this potential friction, some of these unofficial clients have been very popular. Both Facebook and Twitter now offer their own official lightweight clients such as *Facebook Lite*,⁷² *Messenger Lite*,⁷³ and *Twitter Lite*⁷⁴ and cater explicitly to developing countries and their emerging economies. Most of these apps only exist for the Android platform and not for iOS as they cater to users in 'emerging markets', who typically own much more affordable, lower-end Android devices instead of more expensive iPhones.⁷⁵ Apps that fall into this category help negotiate infrastructural limits that communities of practice face, such as limited electrical power supply, slower device replacement cycles, and lower network bandwidth.

Similarly, we find that these four social media platforms are not currently optimised for smartwatch and television integrations, as we find a very small number of such integrations.⁷⁶ Here, too, the limitations of the device or infrastructure are negotiated by the developers of third-party apps. The apps in this category broaden the user base of platforms by catering to new communities and intensify user engagement by working around infrastructural constraints.

Revival

The third group of apps seek to revive formerly supported and discontinued functionality. For example, in the case of Twitter, some third-party apps have revived previous use practices and ways of being on the platform that no longer exists within the official Twitter client. *Twitterific*⁷⁷ brings back Twitter's reverse-chronological timeline, which was controversially replaced with an algorithmically-sorted timeline in 2016 to increase user engagement.⁷⁸ *Twitterific* presents this revival as the acknowledgement of existing use practices for users who want to view all tweets 'in the way you expect' and for users who want to return to a less commercialised version of Twitter by promising a 'clutter free' experience of the reverse-chronological timeline, which is free of promoted tweets and ads.⁷⁹ To achieve this reverse-chronological order, the app made use of functionality available through Twitter's APIs (now deprecated).⁸⁰ Another Twitter client, *Tweecha Prime*⁸¹ revived a former retweet practice of putting 'RT' in front of a tweet to signify a retweet⁸² and *Speed Social for Twitter*⁸³ reversed Twitter's design decision to replace favourites with likes in 2015. The latter app allegedly accommodates users who want their 'stars back' and enables them to 'Use favourite (★) instead of like (♥)'. Crucially, such apps only change the appearance of like buttons back to the former favourite buttons and do not actually modify or revive any of the former action grammars. Such an intervention merely modifies the perceived affordance for users, which may then influence the actual use of that functionality. Hence, the data they produce may introduce different interpretations to the newly established action grammars.

Additionally, some app developers reunite divided families of apps. These apps have been broken up into groups of related apps by the same developer and thus distribute functionality across multiple apps.⁸⁴ *Floating Lite for Facebook*⁸⁵ and *Messenger for Facebook – Security Lock*,⁸⁶ for example, both allow users to reunite Facebook and Messenger back into one single app. This approach is in contrast to apps such as WeChat which offer the entire functionality within a single app. Note that both Google Play and the App Store now accommodate

these broken families by serving related apps of the same developers, which makes it easier to find these 'orphaned' apps. These cases of revival elicit nostalgia among users about discontinued platform-related practices and features. As such, they negotiate between established user practices and platform design decisions – a process that can but does not need to align with platforms' own interests.

Instruction

The fourth group of apps concerns learning about existing platform functionality and use affordances. These apps range from guides for Facebook and Instagram for users interested in deleting their accounts to beginners' and how-to guides for using hashtags and saving videos found on Twitter. Most of these instruction apps, however, appear for Snapchat and Twitter and are guides or tutorials for using specific cameras and lenses or finding friends but also for growing audiences, operating multiple accounts, and running campaigns (again targeting influencers). Furthermore, we find guides for learning how to use Snapchat, especially for understanding its specific terminology and basic interface functionality; this proves that for some users, Snapchat's interface is not intuitive and indeed needs explanation. In 2018, Snapchat introduced a controversial overhaul of its interface that was intended to make the app 'simpler and easier to use, especially for older users'.⁸⁷ However, in doing so, it only made the app even more difficult to use, according to over 1.2 million Snapchat users who signed a petition to urge a reversal of the major redesign.⁸⁸ Similar to revival apps, these instruction apps point to the messy, complicated relationship between, on the one hand, official apps and their changing interface design and functionality and, on the other hand, diverse use practices and user communities.

Transformation

The final group of apps seek to introduce or transform – either implicitly or explicitly – existing action grammars or use practices. These apps can stretch or circumvent the limitations of existing action grammars to introduce new functionality, such as replaying Stories, creating 'tweetstorms', downloading images and videos, saving ephemeral content such as Snaps, muting users on Twitter, hiding users' online status, or monitoring how other users have engaged with or visited a user's profile. This type of functionality is usually not officially supported but may nonetheless rely on official APIs. Out of all forms of regramming, apps in this category interfere most explicitly with platforms' existing action grammars.

First, many apps 'chain' existing action grammars to create workarounds, for instance, to create new grammars. *Regrann*⁸⁹ and *Regram*⁹⁰ both introduce reposting by chaining existing grammars one after another. These reposting apps are then used alongside the official Instagram client, rather than taking its place. That is, both apps are linked and necessary to achieve the functionality: users copy a publicly accessible image URL from Instagram and the third-party app then downloads it onto the device, prepares a repost with a watermark or photo credit, and posts it back to Instagram using the official app. Since it is not possible to post directly to Instagram, developers create workarounds, for example by using a 'share to' window that opens into Instagram. Other apps such as *InRepost*⁹¹ work around this problem by using Instagram's private APIs (i.instagram.com), loading the entire feed and adding a reposting button. Similarly, *Twitterific*⁹² introduces functionality to edit previously posted tweets by chaining existing grammars, which is not officially supported either. As a workaround, it 'immediately deletes your original tweet and redisplay the compose screen with the text of your tweet already filled in'.⁹³ In addition to posting and editing, many apps introduce functionality for downloading or saving content. Snaps and Stories are only available for only a limited amount of time. While users can replay Snaps once, there is no way to save or download them for later use; in fact, saving is actively discouraged and the official client alerts users when others create screenshots. Consequently, apps such as *Snitchchat*,⁹⁴ *Snapsaver*,⁹⁵ and *Record snap story without being detected*⁹⁶ introduce workarounds to save or record Snapchat content without alerting other users. Additionally, while the social media platforms all offer functionality for live broadcasting and video content streaming, none of them supports the official functionality to download or save content (probably due to copyright restrictions). As a workaround, most of these apps rely on APIs to authenticate and download the content concerned (graph.facebook.com, api.twitter.com, stream.twitter.com, i.instagram.com).

Second, in addition to such workarounds, there are apps that promise new functionality but then fail to deliver on their promise. For example, activity and profile monitoring apps such as *Who Viewed My Profile*⁹⁷ and *Who Viewed My Facebook Profile*⁹⁸ claim to give more insights into user engagement when that is not supported. Some of them use the official APIs to retrieve friends or follower lists although the endpoints used do not provide any information about who visited users' profiles (due to privacy protection). As a result, these apps use workarounds that give users the illusion of functionality (and their developers confirm this with disclaimers in their apps' descriptions).⁹⁹

Finally, some apps repurpose existing functionality to introduce alternative use practices or platform imaginaries. The most prominent case is the repurposing of existing Facebook and Snapchat functionality to accommodate dating use practices. The implementations range from gender-based username search functionality (e.g., *Find Girls Username*¹⁰⁰) to offering emojis and GIFs for romantic chatting (e.g., *Between*¹⁰¹) to offering profile picture-based matching functionality (e.g., *AddMe*¹⁰²). Such apps are in tension with platforms' policies and terms of use: while creative and innovative contributions are invited, these policies also require that the overall platform experience be maintained and not altered.¹⁰³

Practices on the Peripheries of Platforms

In this final section, we reflect on the conceptual and methodological implications of our analysis of the various kinds of relations between apps and platforms that we found based on our app-centric approach. We took stock of Android and iOS apps related to four leading social media platforms and used Google Play and Apple's App Store as devices to demarcate our source sets. We found that the infrastructural relations between apps and platforms are not necessarily API-based and traceable but may exist beyond the purview of platforms' governance and control mechanisms – a key focus point in both media and management and information systems research.¹⁰⁴ As such, our app-centric approach provides insights into the marginal and contingent practices of users and developers on the peripheries of platforms.

Like app developers, researchers need to consider the multiple layers of infrastructure supporting apps, especially between apps and app stores (Google Play, App Store), apps and mobile platforms (Android, iOS), and apps and social media platforms (Facebook, Instagram, Snapchat, Twitter). Although the primary focus of our empirical analysis has been on the relations between mobile apps and social media platforms, our methodological approach relies on all three infrastructure layers. For one, app stores have different policies and terms of service (relating to safety, business, performance, design, and legal aspects) that influence the kinds of apps listed, the contents and functionality they offer, and the practices they support or extend. App stores also have their own specific ways of ordering apps and their relations such as through recommender systems that associate apps by the same developers or by topic (in our case social media platform names). When we employ app stores to demarcate source sets for us, we therefore also considered how their specificities influence app results, relevance, and relatedness. Furthermore, we

noticed that both app stores are rather volatile. Within our corpus, 3.95% of Android apps and 6.79% of iOS apps were removed within one month after data collection.¹⁰⁵ These removals were due to general app store maintenance efforts but also due to targeted app removals, including apps that violate social media platforms' policies and terms of service. As we have seen, app stores appeal to their regulatory power by removing apps that violate social media platform policies and terms of service.¹⁰⁶ Finally, development affordances depend not only on the specificities of app stores but also on the mobile platforms these apps are built for. Certain functionalities, such as lock screen customisation and theming (of which we found many in our source sets), are simply available only for Android and not for iOS, for example.

By taking the apps listed in app stores as the starting point of analysis, we looked at the peripheries of social media platforms to identify app–platform relations beyond API-based connections only. As such, we conceived of infrastructure in relational terms as emerging at the interplay between the development of technical or material elements and communities of practice.¹⁰⁷ In addition to API-based infrastructural relations, we distinguished platform-specific practices and features that reveal how social media platforms are used via and alongside third-party clients and apps. These relations are enabled by the official resources that social media platforms offer to third-party developers (e.g., APIs, SDKs), but we also found that many app developers use unofficial and alternative resources to build apps, such as repurposing mobile resources for lightweight functionality. Other apps do not rely on any platform APIs and are designed as complementary apps to be used alongside the official apps. In fact, for all platforms, we found workarounds involving different strategies to resist or overcome infrastructure constraints, demonstrating the creativity of app developers and the marginal status of their apps. Additionally, due to such developer workarounds, not all practices supported by third-party apps are captured by social media platforms. When apps work around official resources, the practices they support may escape data capture because there is not always a pre-structured action grammar underpinning users' activities. This marks another consequence of apps operating on the peripheries of platforms and may well be a motivation for social media platforms to adopt the more popular, newly introduced functionality from third-party apps. In short, we make two contributions to social media and platform studies.¹⁰⁸ The first relates to discussions about the central role of APIs in enabling platforms' programmability by identifying forms of regramming beyond APIs; the second relates to discussions about the capture of users' online activities and behaviours by arguing that not all forms of regramming align with platforms'

existing action grammars or interests.

Analysing platform-related practices and grammars through third-party apps allowed us to explore how app developers engage with social media platforms. As we argued, app developers have different ways of engaging with the specific action grammars offered by platforms. The majority of apps mostly make minor adjustments to accommodate the 'countless needs and niches' of platforms,¹⁰⁹ which we largely consider intensifications and reductions of existing functionality, the use of which feeds back into and aligns well with the logic and grammars of the core platform. With reductions and revivals, the apps also accommodate use practices that can be seen as forms of customisation rather than circumvention – that is, they are mostly aesthetic changes that do not change the fundamental use of the platform. This insight allows revisiting claims around platform-based innovation ecosystems that stress app developers' capacities to envision and produce complementary, value-adding functionality.¹¹⁰ Our analysis of four leading social media platform ecosystems has provided an analytical language to consider such contributions and determine how they actually relate or add value to platforms. In our case, apps could be conceived for example as adding value not by introducing new functionality but by increasing the volume and type of content or engagement via new use practices. Additionally, we found apps that introduce new or transform existing functionality. Sometimes their innovations are adopted by the core platform (e.g., reposting and anti-addiction apps for Instagram, lightweight clients and dating apps for Facebook). The crucial benefit to the social media platform is not necessarily the new functionality itself, which could arguably have been introduced far sooner, but rather the fact that the practice is already established and adopted within the community, thus making the introduction of the new functionality much less risky for the platform while still accommodating the needs and niches of its users.

Contrary to Agre's original formulation of the grammars of action concept, which originated in the context of automating organisational routines and practices, our analysis suggests the need for a more distributed,¹¹¹ layered and platform-specific account of action grammars. When core platforms introduce new features it is not necessarily the case that use practices adapt to those new features (e.g., Twitter's reverse-chronological timeline or favourite button). Moreover, when users or developers come up with alternative ways of using existing action grammars, the activities captured through them may no longer be commensurable (e.g., when Instagram users' original content posts become reposts). While post counts or follower counts remain the same metrics, their

uses and meanings are marginally contested or negotiated. Furthermore, users may chain multiple action grammars within one use practice such as with tweet editing or reposting, which respectively combine delete-and-post or share-and-post functionality, thereby assigning different meanings to these action grammars. Therefore, the development of new action grammars not only is about technological development on the side of platforms but also involves the establishment of support – or infrastructure – for use practices that are currently not (or not officially) supported. As we have argued, that process is very much an interplay between platforms and other stakeholders, each with their own interests.

Conclusion

The main aim of this article was to explore the relations and interactions between third-party apps and social media platforms. We developed the notion of regramming to describe the different kinds of app–platform relations that we derived from our empirical analysis of platform-related practices and features. In particular, we advanced an app-centric methodological approach for identifying the infrastructural relations between apps, social media platforms, app stores, and mobile platforms, as enacted through the practices of both users and developers. This perspective revealed a complex web of infrastructural relations between apps and platforms characterised by the use of official platform resources and by developer workarounds and infrastructure constraints. The forms of regramming that we found are indicative of how – and how much – app developers can negotiate the infrastructural affordances and constraints determined by platforms. We, therefore, encourage future critical research in this area to continue exploring platforms’ peripheries and consider the creative potential of app developers in creating alternative practices and features. Such a perspective should be of particular interest to scholars in the fields of software and platform studies, which have long promoted diversity in entry points, methods of analysis, and descriptions for purposes of critical enquiry into the objects and practices of computational culture. Conceiving platforms and apps in relational and infrastructural terms foregrounds the messiness and contingency of app development on the periphery of platforms.

Acknowledgements

Authors are listed in alphabetical order. Thanks to Emile den Tex and Erik Borra (Digital Methods Initiative, University of Amsterdam) for their development and maintenance of digital methods tools. Thanks also to the *Writing with Networks*

workshop participants (University of Amsterdam) and to the reviewers for their constructive comments.

Funding

Parts of this work were supported by the German Research Foundation (DFG) under Grant DFG-SFB-1187; and the Netherlands Organisation for Scientific Research (NWO) under Grants 275-45-005 and 275-45-009.

Bibliography

Agre, Philip E. "Surveillance and Capture: Two Models of Privacy." *The Information Society* 10, no. 2 (1994): 101–127.
<https://doi.org/10.1080/01972243.1994.9960162>.

Alter, Steven. "Theory of Workarounds." *Communications of the Association for Information Systems* 34 (2014): 1041–1066.

Andreessen, Marc. "The Three Kinds of Platforms You Meet on the Internet." *blog.pmarca.com*, September 16, 2007.
https://pmarchive.com/three_kinds_of_platforms_you_meet_on_the_internet.html.

Apple Developer. "Optimizing for App Store Search." *Apple Developer*.
<https://developer.apple.com/app-store/search/>.

Baldwin, Carliss Young, and C. Jason Woodard. "The Architecture of Platforms: A Unified View." In *Platforms, Markets and Innovation*, edited by Annabelle Gawer, 19–44. Cheltenham, UK: Edward Elgar Publishing, 2009.

Blackwell, Courtney, Jeremy Birnholtz, and Charles Abbott. "Seeing and Being Seen: Co-Situation and Impression Formation Using Grindr, a Location-Aware Gay Dating App." *New Media & Society* 17, no. 7 (2015): 1117–1136.
<https://doi.org/10.1177/1461444814521595>.

Bodle, Robert. "Regimes of Sharing." *Information, Communication & Society* 14, no. 3 (2011): 320–337. <https://doi.org/10.1080/1369118X.2010.542825>.

Bogost, Ian, and Nick Montfort. "Platform Studies: Frequently Questioned Answers." In *Proceedings of the Digital Arts and Culture Conference*. University

of California, Irvine, 2009.

Bucher, Taina. "Objects of Intense Feeling: The Case of the Twitter API." *Computational Culture* 3 (2013).

<http://computationalculture.net/article/objects-of-intense-feeling-the-case-of-the-twitter-api>.

Bucher, Taina, and Anne Helmond. "The Affordances of Social Media Platforms." In *The SAGE Handbook of Social Media*, edited by Jean Burgess, Thomas Poell, and Alice Marwick, 233–53. London: SAGE Publications, 2018.

Constantinides, Panos, Ola Henfridsson, and Geoffrey G. Parker. "Introduction—Platforms and Infrastructures in the Digital Age." *Information Systems Research* 29, no. 2 (2018): 381–400. <https://doi.org/10.1287/isre.2018.0794>.

Constine, Josh. "Instagram Now Lets You Regram Your Posts to Multiple Accounts." *TechCrunch*, January 9, 2019.

<http://social.techcrunch.com/2019/01/09/instagram-self-regram/>.

———. "Twitter Cuts off API Access to Follow/Unfollow Spam Dealers."

TechCrunch, January 31, 2019. <http://social.techcrunch.com/2019/01/31/dont-buy-twitter-followers/>.

Dieter, Michael, Carolin Gerlitz, Anne Helmond, Nathaniel Tkacz, Fernando N. van der Vlist, and Esther Weltevrede. "Multi-Situated App Studies: Methods and Propositions." *Social Media + Society* 5, no. 2 (2019): 1–15.

<https://doi.org/10.1177/2056305119846486>.

Duguay, Stefanie. "Dressing up Cinderella: Interrogating Authenticity Claims on the Mobile Dating App Tinder." *Information, Communication & Society* 20, no. 3 (2017): 1–17. <https://doi.org/10.1080/1369118X.2016.1168471>.

Eaton, Ben, Silvia Elaluf-Calderwood, Carsten Sorensen, and Youngjin Yoo. "Distributed Tuning of Boundary Resources: The Case of Apple's iOS Service System." *MIS Quarterly* 39, no. 1 (2015): 217–243.

<https://doi.org/10.25300/MISQ/2015/39.1.10>.

Evans, Peter C., and Rahul C. Basole. "Revealing the API Ecosystem and Enterprise Strategy via Visual Analytics." *Communications of the ACM* 59, no. 2 (2016): 26–28. <https://doi.org/10.1145/2856447>.

Facebook for Developers. "Android SDK." *Facebook for Developers*.
<https://developers.facebook.com/docs/android>.

———. "APIs and SDKs." *Facebook for Developers*.
<https://developers.facebook.com/docs/apis-and-sdks/>.

———. "Instagram Graph API." *Facebook for Developers*.
<https://developers.facebook.com/docs/instagram-api>.

———. "Platform Policy." *Facebook for Developers*.
<https://developers.facebook.com/policy/>.

Fuller, Matthew. "Software Studies." In *The Johns Hopkins Guide to Digital Media*, edited by Marie-Laure Ryan, Lori Emerson, and Benjamin J. Robertson, 463–466. Baltimore, MD: Johns Hopkins University Press, 2014.

Gawer, Annabelle, and Michael A. Cusumano. "Industry Platforms and Ecosystem Innovation." *Journal of Product Innovation Management* 31, no. 3 (2014): 417–33. <https://doi.org/10.1111/jpim.12105>.

Gehl, Robert. *Reverse Engineering Social Media: Software, Culture, and Political Economy in New Media Capitalism*. Philadelphia, PA: Temple University Press, 2014.

Gerlitz, Carolin, and Anne Helmond. "The Like Economy: Social Buttons and the Data-Intensive Web." *New Media & Society* 15, no. 8 (2013): 1348–1365. <https://doi.org/10.1177/1461444812472322>.

Gerlitz, Carolin, and Bernhard Rieder. "Tweets Are Not Created Equal. A Platform Perspective on Social Media Metrics." *International Journal of Communication* 12 (2018): 528–547.

Gerlitz, Carolin, Fernando N. van der Vlist, Anne Helmond, and Esther Weltevrede. "App Support Ecologies: An Empirical Investigation of App–Platform Relations." In *1st 2016 Annual Conference of the CRC 'Media of Cooperation'*. Artur-Woll-Haus, University of Siegen, Germany, December 2016. <https://bit.ly/app-support-ecologies>.

Gerlitz, Carolin, and Esther Weltevrede. "What Happens to ANT, and Its

Emphasis on the Socio-Material Grounding of the Social, in Digital Sociology?" In *The Routledge Companion to Actor-Network Theory*, edited by Anders Blok, Ignacio Farías, and Celia Roberts. London, UK and New York, US: Routledge.

Ghazawneh, Ahmad, and Ola Henfridsson. "Balancing Platform Control and External Contribution in Third-Party Development: The Boundary Resources Model." *Information Systems Journal* 23, no. 2 (March 1, 2013): 173–192. <https://doi.org/10.1111/j.1365-2575.2012.00406.x>.

Gillespie, Tarleton. "The Politics of 'Platforms.'" *New Media & Society* 12, no. 3 (2010): 347–364. <https://doi.org/10.1177/1461444809342738>.

Google Play Console. "Google Play Console." *Google Play Console*. <https://play.google.com/apps/publish/>.

Google Play Console Help. "Get Discovered on Google Play Search." *Google Play Console Help*. <https://support.google.com/googleplay/android-developer/answer/4448378>.

Helmond, Anne. "The Platformization of the Web: Making Web Data Platform Ready." *Social Media + Society* 1, no. 2 (July 1, 2015): 1–11. <https://doi.org/10.1177/2056305115603080>.

Instagram. "New: Share Feed Posts to Stories." *Instagram*, May 17, 2018. <https://instagram-press.com/blog/2018/05/17/new-share-feed-posts-to-stories/>.

———. "Platform Policy." *Instagram*. <https://www.instagram.com/about/legal/terms/api/>.

———. "Thinking Outside the Square: Support for Landscape and Portrait Formats on Instagram." *Instagram Info Center*, August 27, 2015. <https://instagram-press.com/blog/2015/08/27/thinking-outside-the-square-support-for-landscape-and-portrait-formats-on-instagram/>.

Instagram Info Center. "Reducing Inauthentic Activity on Instagram." *Instagram Info Center*, November 19, 2018. <https://instagram-press.com/blog/2018/11/19/reducing-inauthentic-activity-on-instagram/>.

John, Nicholas A. *The Age of Sharing*. Malden, MA: Polity Press, 2017.

Langlois, Ganaele, and Greg Elmer. "The Research Politics of Social Media Platforms." *Culture Machine* 14 (2013): 1–17.

Nieborg, David B., and Anne Helmond. "The Political Economy of Facebook's Platformization in the Mobile Ecosystem: Facebook Messenger as a Platform Instance." *Media, Culture & Society* 41, no. 2 (2019): 196–218.
<https://doi.org/10.1177/0163443718818384>.

Paßmann, Johannes, and Carolin Gerlitz. ",Good` Platform-Political Reasons for ,bad` Platform-Data. Zur Sozio-Technischen Geschichte Der Plattformaktivitäten Fav, Retweet Und Like." *Mediale Kontrolle Unter Beobachtung* 3, no. 1 (2014): 1–40.

Perez, Sarah. "6 Million Users Had Installed Third-Party Twitter Clients." *TechCrunch*, August 17, 2018. <http://social.techcrunch.com/2018/08/17/6-million-users-had-installed-third-party-twitter-clients/>.

———. "Google Follows in Apple's Footsteps by Cleaning up Its Play Store." *TechCrunch*, July 27, 2018. <http://social.techcrunch.com/2018/07/27/google-follows-in-apples-footsteps-by-cleaning-up-its-play-store/>.

Pinch, Trevor J., and Wiebe E. Bijker. "The Social Construction of Facts and Artefacts: Or How the Sociology of Science and the Sociology of Technology Might Benefit Each Other." *Social Studies of Science* 14, no. 3 (1984): 399–441.
<https://doi.org/10.1177/030631284014003004>.

Plantin, Jean-Christophe, Carl Lagoze, Paul N. Edwards, and Christian Sandvig. "Infrastructure Studies Meet Platform Studies in the Age of Google and Facebook." *New Media & Society* 20, no. 1 (2018): 293–310.
<https://doi.org/10.1177/1461444816661553>.

[Port:9]. "Why Regram?" *[Port:9]*, November 6, 2012.
<http://www.port9.com/post/44691426770/why-regram>.

Puschmann, Cornelius, and Jean Burgess. "The Politics of Twitter Data." In *Twitter and Society*, edited by Katrin Weller, Axel Bruns, Jean Burgess, Merja Mahrt, and Cornelius Puschmann, 43–54. New York, NY: Peter Lang, 2014.

de Reuver, Mark, Carsten Sørensen, and Rahul C. Basole (2018) "The Digital

Platform: A Research Agenda." *Journal of Information Technology* 33(2): 124–135. <https://doi.org/10.1057/s41265-016-0033-3>.

Rogers, Richard. *Digital Methods*. Cambridge, MA: The MIT Press, 2013.

Roy, Gautam. "How We Built Facebook Lite for Every Android Phone and Network." *Facebook Code*, March 9, 2016. <https://code.fb.com/android/how-we-built-facebook-lite-for-every-android-phone-and-network/>.

Snap Business. "Apply to the Snapchat Partners Program." *Snap Business*. <https://forbusiness.snapchat.com/partners/apply>.

———. "Introducing Snap Kit." *Snap Business*, June 14, 2018. <https://forbusiness.snapchat.com/blog/introducing-snap-kit/>.

Snap Inc. "Third-Party Applications and the Snapchat API." *Snap Inc. News*, October 14, 2014. <https://www.snap.com/en-US/news/post/third-party-applications-and-the-snapchat-api>.

Snapchat Support. "Third-Party Applications and Plugins." *Snapchat Support*. <https://support.snapchat.com/en-US/a/third-party>.

Star, Susan Leigh, and Karen Ruhleder. "Steps Toward an Ecology of Infrastructure: Design and Access for Large Information Spaces." *Information Systems Research* 7, no. 1 (1996): 111–134. <https://doi.org/10.1287/isre.7.1.111>.

Statista. "Global Social Media Ranking 2018." *Statista*, July 2018. <https://www.statista.com/statistics/272014/global-social-networks-ranked-by-number-of-users/>.

Tiwana, Amrit. *Platform Ecosystems: Aligning Architecture, Governance, and Strategy*. Waltham, MA: Morgan Kaufmann, 2014.

Twitter Developer. "Build Mobile Apps with Twitter Kit." *Twitter Developer*. <https://developer.twitter.com/en/docs/publisher-tools/twitterkit/overview.html>.

———. "Developer Agreement and Policy." *Twitter Developer*, May 25, 2018. <https://developer.twitter.com/en/developer-terms/agreement-and-policy.html>.

———. “Developer Policy.” *Twitter Developer*, November 3, 2017. <https://developer.twitter.com/en/developer-terms/policy.html>.

———. “Twitter Developer Platform.” *Twitter Developer*. <https://developer.twitter.com/content/developer-twitter/en.html>.

Twitter Help Center. “Automation Rules.” *Twitter Help Center*, November 3, 2017. <https://help.twitter.com/en/rules-and-policies/twitter-automation>.

Twitterrific. “Editing Tweets in Twitterrific (iOS and macOS).” *The Iconfactory*. <https://support.iconfactory.com/kb/twitterrific/editing-tweets-in-twitterrific-ios-and-macos>.

van Dijck, José. *The Culture of Connectivity: A Critical History of Social Media*. Oxford, UK: Oxford University Press, 2013.

van der Vlist, Fernando N. “Accounting for the Social: Investigating Commensuration and Big Data Practices at Facebook.” *Big Data & Society* 3, no. 1 (2016): 1–16. <https://doi.org/10.1177/2053951716631365>.

Wagner, Kurt. “Instagram Is Limiting How Much Data Some Developers Can Collect from Its API — and Cutting off Others Altogether.” *Recode*, April 2, 2018. <https://www.recode.net/2018/4/2/17189512/instagram-api-facebook-cambridge-analytica>.

Werning, Stefan. “‘Re-Appropriating’ Facebook? Web API Mashups as Collective Cultural Practice.” *Digital Culture & Society* 3, no. 2 (2017): 183–204. <https://doi.org/10.14361/dcs-2017-0211>.

Wilson, Fred. “App Constellations.” *AVC*, May 18, 2014. <https://avc.com/2014/05/app-constellations/>.

Zaidi, Deena. “Snap’s Redesign Faces Backlash From Loyalists.” *Seeking Alpha*, February 22, 2018. <https://seekingalpha.com/article/4149405-snaps-redesign-faces-backlash-loyalists>.

Author Biographies

Carolyn Gerlitz is a professor of Digital Media and Methods at the University of Siegen. Her research interests include digital media technologies, quantification,

inventive methods, software studies, platforms, sensor media, and critical data studies.

Anne Helmond is an assistant professor of New Media and Digital Culture at the University of Amsterdam. Her research interests include software studies, platform studies, app studies, digital methods, and web history.

Fernando van der Vlist is a PhD candidate at Utrecht University and a research associate with the Collaborative Research Centre "Media of Cooperation" at the University of Siegen. His research interests include software studies, digital methods, social media and platform studies, app studies, and critical data studies.

Esther Weltevrede is an assistant professor of New Media and Digital Culture at the University of Amsterdam. Her research interests include digital methods, software and platform studies, app studies, data infrastructures, and social media automation.

Notes

1. Instagram, "New: Share Feed Posts to Stories." ↩
2. Khader, *Regram (Repost Photo & Video for Instagram)*, <https://play.google.com/store/apps/details?id=regram.instagram.download>. ↩
3. [Port:9 ↩
4. As of January 2019, Instagram has started to roll out a new "self regram" feature, allowing users to cross-syndicate their own content to multiple Instagram accounts. Constine, "Instagram Now Lets You Regram Your Posts to Multiple Accounts." ↩
5. Perez, "6 Million Users Had Installed Third-Party Twitter Clients." ↩
6. For example, as of August 2018, IFTTT lists 163 Instagram Applets while Google Play returns 250 app search query results for [Instagram ↩
7. Statista, "Global Social Media Ranking 2018." ↩
8. Agre, "Surveillance and Capture." ↩
9. Bogost and Montfort, "Platform Studies"; Gillespie, "The Politics of 'Platforms'"; Langlois and Elmer, "The Research Politics of Social Media Platforms"; van Dijck, *The Culture of Connectivity*. ↩
10. Cf., Fuller, "Software Studies," 463. ↩
11. Bodle, "Regimes of Sharing"; Bucher, "Objects of Intense Feeling"; Gerlitz and Helmond, "The Like Economy"; Gillespie, "The Politics of 'Platforms'";

- Puschmann and Burgess, "The Politics of Twitter Data.", Werning, "'Re-Appropriating' Facebook?" ↵
12. Plantin et al., "Infrastructure Studies Meet Platform Studies in the Age of Google and Facebook." ↵
 13. Nieborg and Helmond, "The Political Economy of Facebook's Platformization in the Mobile Ecosystem." ↵
 14. Helmond, "The Platformization of the Web." ↵
 15. Andreessen, "The Three Kinds of Platforms You Meet on the Internet." ↵
 16. Facebook for Developers, "Platform Policy." ↵
 17. Twitter Developer, "Developer Policy." ↵
 18. Snap Kit Docs, "Snap Kit Documentation." ↵
 19. Pinch and Bijker, "The Social Construction of Facts and Artefacts"; cf. Paßmann and Gerlitz, ",Good' Platform-Political Reasons for ,Bad' Platform-Data." ↵
 20. Gerlitz et al., "App Support Ecologies." ↵
 21. Constantinides, Henfridsson, and Parker, "Platforms and Infrastructures in the Digital Age"; de Reuver, Sørensen, and Basole, "The Digital Platform"; Tiwana, *Platform Ecosystems*. ↵
 22. Baldwin and Woodard, "The Architecture of Platforms"; Gawer and Cusumano, "Industry Platforms and Ecosystem Innovation." ↵
 23. de Reuver, Sørensen, and Basole, "The Digital Platform." ↵
 24. Eaton et al., "Distributed Tuning of Boundary Resources"; Ghazawneh and Henfridsson, "Balancing Platform Control and External Contribution in Third-Party Development." ↵
 25. Tiwana, *Platform Ecosystems*, 5–6. ↵
 26. Dieter et al., "Multi-Situated App Studies"; Gerlitz et al., "App Support Ecologies." ↵
 27. Evans and Basole, "Revealing the API Ecosystem and Enterprise Strategy via Visual Analytics"; Werning, "'Re-Appropriating' Facebook?"; ProgrammableWeb, "API Directory," <https://www.programmableweb.com/apis/directory>; IFTTT, "Services," <https://ifttt.com/services>. ↵
 28. Star and Ruhleder, "Steps Toward an Ecology of Infrastructure," 113, emphasis in original. ↵
 29. Ibid., 113. ↵
 30. Cf. Agre, "Surveillance and Capture," ↵
 31. Ibid., 107–109. ↵
 32. Gerlitz and Rieder, "Tweets Are Not Created Equal"; van der Vlist, "Accounting for the Social." ↵
 33. van Dijck, *The Culture of Connectivity*. ↵

34. Cf., Andreessen, "The Three Kinds of Platforms You Meet on the Internet." ↩
35. Dieter et al., "Multi-Situated App Studies." ↩
36. Statista, "Global Social Media Ranking 2018." ↩
37. Dieter et al., "Multi-Situated App Studies"; Rogers, *Digital Methods*, 118. ↩
38. Apple Developer, "Optimizing for App Store Search"; Google Play Console, "Google Play Console"; Google Play Console Help, "Get Discovered on Google Play Search." ↩
39. Dieter et al., "Multi-Situated App Studies." ↩
40. Data collected in mid-July 2018. ↩
41. Digital Methods Initiative, *Google Play 'Similar' Apps*, <https://tools.digitalmethods.net/beta/googlePlaySimilar/>; *ibid.*, *iTunes Store*, <https://tools.digitalmethods.net/beta/itunesStore/>. ↩
42. Most apps in our corpus had English descriptions (95.71%), followed by a relatively small number in Arabic (1.14%), and Spanish (0.91%), which led us to include some search patterns in these languages. All searches were processed computationally with 'grep', a UNIX-based command-line utility for searching textual data. Each practice and feature is searched for with multiple search patterns (although we counted positive matches only once per app). ↩
43. Blackwell, Birnholtz, and Abbott, "Seeing and Being Seen"; Duguay, "Dressing up Cinderella." ↩
44. This is after the company discovered that 'nearly one in five photos or videos people post aren't in the square format', which suggests that these photos or videos are created with third-party apps before they are posted on Instagram. Instagram, "Thinking Outside the Square." ↩
45. John, *The Age of Sharing*. ↩
46. Facebook for Developers, "Instagram Graph API." ↩
47. Snap Business, "Apply to the Snapchat Partners Program." ↩
48. Dieter et al., "Multi-Situated App Studies." ↩
49. Facebook for Developers, "Android SDK." ↩
50. Twitter Developer, "Build Mobile Apps with Twitter Kit." ↩
51. Snap Inc., "Third-Party Applications and the Snapchat API." ↩
52. Alter, "Theory of Workarounds." ↩
53. Facebook for Developers, "APIs and SDKs"; Snap Inc., "Third-Party Applications and the Snapchat API." ↩
54. Constine, "Twitter Cuts off API Access to Follow/Unfollow Spam Dealers"; Snapchat Support, "Third-Party Applications and Plugins"; Wagner, "Instagram Is Limiting How Much Data Some Developers Can Collect from Its API — and Cutting off Others Altogether." ↩
55. Gehl, *Reverse Engineering Social Media*, 81. ↩

56. Neelkanth, *Beautify – Skin Tone Filters*,
<https://itunes.apple.com/us/app/beautify-skin-tone-filters/id1136143080>. ↩
57. Google, *Snapseed*, <https://itunes.apple.com/us/app/snapseed/id439438619>.
↩
58. Gerlitz and Weltevrede, "What Happens to ANT, and Its Emphasis on the Socio-Material Grounding of the Social, in Digital Sociology?" ↩
59. Hootsuite Media, *Hootsuite – Social Media Tools*,
<https://itunes.apple.com/us/app/hootsuite-social-media-tools/id341249709>.
↩
60. Everypost, *Everypost for Social Media*,
<https://itunes.apple.com/us/app/everypost-for-social-media/id572530903>.
↩
61. Twitter Developer, "Developer Policy"; Twitter Help Center, "Automation Rules." ↩
62. Constine, "Twitter Cuts off API Access to Follow/Unfollow Spam Dealers." ↩
63. Statusbrew, *Statusbrew: Social Media Tools*,
<https://itunes.apple.com/us/app/statusbrew-social-media-tools/id1079388184>. ↩
64. Codigami Technologies, *Crowdfire*,
<https://itunes.apple.com/us/app/crowdfire/id528626975>. ↩
65. Constine, "Twitter Cuts off API Access to Follow/Unfollow Spam Dealers." ↩
66. Instagram, "Platform Policy." ↩
67. Twitter Developer, "Twitter Developer Platform." ↩
68. Instagram Info Center, "Reducing Inauthentic Activity on Instagram." ↩
69. sinProject, *Tweecha Lite for Twitter: Presented in Papers*,
<https://play.google.com/store/apps/details?id=net.sinproject.android.tweecha.lite>. ↩
70. Facebook for Developers, "Platform Policy." ↩
71. Twitter Developer, "Developer Policy." ↩
72. Facebook, *Facebook Lite*, <https://play.google.com/store/apps/details?id=com.facebook.lite>. ↩
73. Facebook, *Messenger Lite*, <https://play.google.com/store/apps/details?id=com.facebook.mlite>. ↩
74. Twitter, *Twitter Lite*, <https://play.google.com/store/apps/details?id=com.twitter.android.lite>. ↩
75. Roy, "How We Built Facebook Lite for Every Android Phone and Network." ↩
76. Kumagai, *Looking for Twitter*, <https://play.google.com/store/apps/details?id=com.droibit.looking>. ↩
77. The Iconfactory, *Twitterrific 5 for Twitter*,
<https://itunes.apple.com/us/app/twitterrific-5-for-twitter/id580311103>. ↩

78. Bucher and Helmond, "The Affordances of Social Media Platforms." ↩
79. The Iconfactory, *Twitterrific 5 for Twitter*,
<https://itunes.apple.com/us/app/twitterrific-5-for-twitter/id580311103>. ↩
80. Perez, "6 Million Users Had Installed Third-Party Twitter Clients." ↩
81. sinProject, *Tweecha Prime for Twitter: Presented in papers*,
<https://play.google.com/store/apps/details?id=net.sinproject.android.tweecha.prime>. ↩
82. Cf. Paßmann, in this issue. ↩
83. Shikh Apps, *Speed Social for Twitter*,
<https://play.google.com/store/apps/details?id=net.alshikh.speedtwi>. ↩
84. Wilson, "App Constellations." ↩
85. sarvopari infotech, *Floating Lite for Facebook*,
<https://play.google.com/store/apps/details?id=com.sarvopari.lite.floatingfb>.
↩
86. Rain Studio, *Messenger for Facebook – Security Lock*,
<https://play.google.com/store/apps/details?id=com.a2z.liteforfacebook>. ↩
87. Zaidi, "Snap's Redesign Faces Backlash From Loyalists." ↩
88. Ibid. ↩
89. JaredCo, *Repost for Instagram – ReGrann*,
<https://play.google.com/store/apps/details?id=com.jaredco.regrann>. ↩
90. Khader, *Regram (Repost Photo & Video for Instagram)*,
<https://play.google.com/store/apps/details?id=regram.instagram.download>.
↩
91. BillApps, *InRepost- Repost for Instagram*,
<https://play.google.com/store/apps/details?id=com.billApps.repostForInstagram>. ↩
92. The Iconfactory, *Twitterrific 5 for Twitter*,
<https://itunes.apple.com/us/app/twitterrific-5-for-twitter/id580311103>. ↩
93. *Twitterrific*, "Editing Tweets in Twitterrific (iOS and macOS)." ↩
94. Microcore, *Snitchchat – The Snapchat Screenshot Tool*,
<https://play.google.com/store/apps/details?id=com.studiow.top.snitchchat>.
↩
95. V-Ware, *SnapSaver*, https://play.google.com/store/apps/details?id=com.v_ware.snapsaver. ↩
96. PINEAPPLE, *Record Snap Story Without Being Detected*,
<https://play.google.com/store/apps/details?id=com.dododev.storyrecord>. ↩
97. Loneman Labs, *Who Viewed My Profile? Followers Insight Plus*,
<https://play.google.com/store/apps/details?id=com.lonemanlabs.whoviewedmyprofile.facebook>. ↩

98. Who Viewed My Profile, *Who Viewed My Facebook Profile, Profile Tracker*, <https://play.google.com/store/apps/details?id=com.WhoVisited.My.Facebook.app>. ↩
99. E.g., as the developers of *Who Viewed My Profile* note, 'We do our best to bring you who might have checked your Facebook profile recently; however as Facebook does not disclose who viewed data directly what our app shows you is just a good estimation.' ↩
100. Global Tech Social, *Find Girls Username, Friend for Snapchat Usernames*, https://play.google.com/store/apps/details?id=girls.username.girls_username_for_snapchat. ↩
101. VCNC, *Between – Private Couples App*, <https://play.google.com/store/apps/details?id=kr.co.vcnc.android.couple>. ↩
102. Return Zero, *AddMe – Friends & Usernames for Snapchat & Kik*, <https://play.google.com/store/apps/details?id=co.addme.friends>. ↩
103. Facebook for Developers, "Platform Policy"; Twitter Developer, "Developer Agreement and Policy." ↩
104. Bucher, "Objects of Intense Feeling"; Eaton et al., "Distributed Tuning of Boundary Resources"; Ghazawneh and Henfridsson, "Balancing Platform Control and External Contribution in Third-Party Development"; Puschmann and Burgess, "The Politics of Twitter Data." ↩
105. Perez, "Google Follows in Apple's Footsteps by Cleaning up Its Play Store." ↩
106. Constine, "Twitter Cuts off API Access to Follow/Unfollow Spam Dealers"; Snapchat Support, "Third-Party Applications and Plugins." ↩
107. Star and Ruhleder, "Steps Toward an Ecology of Infrastructure." ↩
108. Bogost and Montfort, "Platform Studies"; Helmond, "The Platformization of the Web"; Langlois and Elmer, "The Research Politics of Social Media Platforms"; Paßmann and Gerlitz, "Good` Platform-Political Reasons for ,Bad` Platform-Data"; Plantin et al., "Infrastructure Studies Meet Platform Studies in the Age of Google and Facebook." ↩
109. Andreessen, "The Three Kinds of Platforms You Meet on the Internet." ↩
110. Cf. Gawer and Cusumano, "Industry Platforms and Ecosystem Innovation." ↩
111. Agre, "Surveillance and Capture." ↩