Original Research Article

Weaving seams with data: Conceptualizing City APIs as elements of infrastructures

Christoph Raetzsch[1], Gabriel Pereira[2], Lasse S Vestergaard[3] and Martin Brynskov[2]

Abstract
This article addresses the role of application programming interfaces (APIs) for integrating data sources in the context of smart cities and communities. On top of the built infrastructures in cities, application programming interfaces allow to weave new kinds of seams from static and dynamic data sources into the urban fabric. Contributing to debates about "urban informatics" and the governance of urban information infrastructures, this article provides a technically informed and critically grounded approach to evaluating APIs as crucial but often overlooked elements within these infrastructures. The conceptualization of what we term City APIs is informed by three perspectives: In the first part, we review established criticisms of proprietary social media APIs and their crucial function in current web architectures. In the second part, we discuss how the design process of APIs defines conventions of data exchanges that also reflect negotiations between API producers and API consumers about affordances and mental models of the underlying computer systems involved. In the third part, we present recent urban data innovation initiatives, especially CitySDK and OrganiCity, to underline the centrality of API design and governance for new kinds of civic and commercial services developed within and for cities. By bridging the fields of criticism, design, and implementation, we argue that City APIs as elements of infrastructures reveal how urban renewal processes become crucial sites of socio-political contestation between data science, technological development, urban management, and civic participation.

Keywords
Application Programming Interface (API), infrastructure, Internet of Things (IoT), interface design, social urban data, smart city

Introduction

In February 2018, Google's (i.e. Alphabet's) Sidewalk Labs announced the launch of *Coord*, a platform for integrated urban mobility that, according to the company's blog, would "unlock a seamless trip experience for people in cities" (Sidewalk Labs, 2018).[1] This seamlessness is possible because "APIs are standardized for the whole industry." The *Coord* platform is framed as a way to "empower cities to create and enforce rules around curb usage" and effectively monitor parking meters, ride shares, and delivery services, at least in San Francisco and New York for the time being. The evocation of seamlessness in this context implies two overlapping developments associated with big data and smart cities. First, by integrating a broad range of different sensors and data sources with real-time analytics in standardized application programming interfaces (APIs), new ways of managing urban messiness become possible. Second, these new ways are realized as services attuned to the specific needs of very different stakeholders—ranging from city officials to delivery companies to taxi and rideshare platforms to

[1]Freie Universität Berlin, Germany
[2]Aarhus University, Denmark
[3]Aarhus University, Denmark; Alexandra Institute, Denmark

Corresponding author:
Christoph Raetzsch, Department of Political and Social Sciences, Institute for Media and Communication Studies, Freie Universität Berlin, Garystr. 55, Berlin 14195, Germany.
Email: christoph.raetzsch@fu-berlin.de

commuters and city travelers. *Coord* is heralded as a universal plug and crucial element for accessing new kinds of urban information infrastructures. This article offers a technically informed and critically grounded approach to evaluating the contradictory aspirations that define APIs as elements of infrastructures for accessing the seams through which cities are perceived and managed. Unobtrusive as APIs are to end users, it is argued that the design and implementation of these interfaces to data represents a "zone of contestation" (Hookway, 2014: 16) between improved public service delivery, new digital markets, and new forms of civic participation.

The promise of seamlessness between data sources insinuated through *Coord* appears as "both the end and the means of large parts of the current digitalized data markets" (Wadmann and Hoeyer, 2018: 2). But achieving this seamlessness is not only a technical challenge. It impacts cities at the organizational level where infrastructures are defined and installed, where data is gathered and analyzed, and where the design of interfaces implements assumptions about who is eligible and able to use them. What kinds of seams get woven with data into the urban fabric through APIs is therefore a technological, political, and social concern at the same time. This article contributes to an expanding body of scholarship on the connections between new urban information infrastructures, big data science and analytics, Internet of Things (IoT), and the new dimensions of public space. This research addresses how cities respond to the proliferation of code and algorithms in understanding the lived reality in city spaces, how sensor networks enable real-time management, and how demands for efficiency and economic innovation are brought together with civic values and new potentials of participation (Kitchin et al., 2017; Kitchin and Perng, 2016). In this interdisciplinary research field, identified respectively as "urban informatics" (Foth et al., 2011), "street computing" (Foth et al., 2015, 2014; Robinson et al., 2012) or "urban science" (Batty, 2013), new modes of governance, management, entrepreneurship, and civic participation are being negotiated and tested.

APIs here emerge as crucial elements of infrastructures that leverage accessibility to new kinds of seams within the urban fabric of data. Looking at APIs from their infrastructural function underlines how infrastructures are "pervasive enabling resources in network form," resources that are always "relational" as they are actualized in daily practices (Bowker et al., 2010: 98, see also Star, 2002; Star and Ruhleder, 1996). For city administrators such infrastructures are essential for providing public services, while for end users they are often taken for granted to achieve everyday tasks. While infrastructure is most readily associated with

durable systems of provisioning basic necessities, distributed global communication networks are increasingly becoming indispensable for localized action and practice (Sandvig, 2013). The study of cities and the study of information systems, to paraphrase Star, thus converge on a common interest in infrastructure as the durable yet contested and often invisible enabler of social life (see Star, 1999: 378).

The competing aspirations around new urban information infrastructures have for some time focused mainly on the massive integration effort of governments and cities to harness only those "rich seams of data that can [be] used to better depict, model and predict urban processes and simulate the likely outcomes of future urban development" (Kitchin, 2014b: 2). Measuring and quantifying the patterns of urban life through data puts forward new forms of "algorhythmic governance" (Coletta and Kitchin, 2017). It also enables interventions in the "realtimeness" of cities (Kitchin, 2018) that satisfy demands for efficient public spending and sustainable living as well as enabling tighter control of messy social processes, e.g. through surveillance of public space or preventive policing. Both kinds of intervention are based on integrating and representing previously solitary, ordinary or invisible "real-time streams" of various kinds of data (Beer, 2011: 143). Deploying sensor networks and CCTV technologies across city spaces epitomizes how "computation is built into the fabric of urban infrastructure" (Coletta and Kitchin, 2017: 1). At the same time, such applications "restructure and recommodify many previously 'public' domains of everyday life and material culture within western capitalist nations" (Graham, 2016: 565). Who has access to data about public life and for what purposes is increasingly a concern for city governments trying to avoid vendor lock-in and long-term dependency on corporate and proprietary platforms operating largely outside of public oversight and regulation.

The overt focus on potentials of surveillance and exclusion in these debates exposes an ongoing struggle between the necessities of city governments to adequately measure and manage vital urban processes and demands by citizens to participate in shaping these processes. On the one hand, urban data can refer to statistical or real-time information about a city's daily flows collected mainly for management and planning purposes by city departments for their respective tasks. On the other hand, urban information infrastructures also open up new sources for understanding cities and the lives of their citizens within a "datafied society" (Van Es and Schäfer, 2017). Psyllidis has coined the term "social urban data" to describe the wider domain of data (and information) derived from communication between individuals and groups about

or within a city through digital media. Such data can be "implicitly-generated social data... sourced from sensors, such as GPS trackers, RFID card readers, and cameras, as well as from mobile phones." But it can also entail communication posted publicly on social media profiles, personal websites, or in communications by associations and citizen groups. Following Psyllidis, social urban data then describes a combination of "process-mediated, machine-generated, and human-sourced" data: official statistics, automatically retrieved data from sensors, and data generated from quotidian interactions of citizens with urban spaces and each other (Psyllidis, 2016: 60f).

The role of APIs in making diverse sources for social urban data available brings up the crucial question how "interfaces could highlight the seams—in our infrastructural networks, between various layers of the urban stack, and even within the social fabric" (Mattern, 2014; cf. Bratton, 2016). The concept of City APIs discussed here is not limited to specific applications akin to Sidewalk's *Coord*—there is not one API that makes an entire city seamlessly accessible. Rather, through the conceptual discussion of APIs in and for cities we address how the weaving of seams with data reflects conflicting aspirations about what purposes new infrastructures shall serve and who benefits in what way. The impression of a seamless integration of data sources in specific use cases rests on a prior negotiation of aspirations and goals that vary greatly between private, commercial, and civic actors as well as public institutions. As an element of infrastructures, an API can serve administrators to control traffic and optimize public services. But an API can also be envisioned as a way to allow citizens to tap into various seams of the social fabric in a city, to reveal submerged layers of histories, experiences, and identities in urban spaces (cf. De Waal, 2014; Mattern, 2017a, 2017b; Sudjic, 2016). In programming terms, an API can be described simply as a "set of routines, protocols, and tools for building software applications" (Forum Virium, 2017: 9). Its role within an infrastructure is to make accessible certain types of data or functions in a standardized way which can be used in a range of subsequent applications. But because an API negotiates between the anticipation of a user (about what kind of resource it offers) and the complexities of a system (how this resource is made available), its role as an element of infrastructure highlights a socio-political contestation over access to data expressed in technological terms.

This paper will discuss APIs as elements of infrastructures from three perspectives. In the first part, we review established criticisms of proprietary social media APIs, pointing out the crucial functions of APIs in current web architectures to connect users and data across applications and platforms. In the second part, we discuss how the design process of APIs defines conventions of data exchanges that also reflect negotiations between API producers and API consumers about affordances and mental models of the underlying computer systems involved. This part will highlight that APIs are far from neutral in this process but embed particular intended uses and assumptions in their design. In the third part, we discuss APIs in the context of open government data (OGD), urban dashboards, and recent urban data initiatives such as the City Service Development Kit (CitySDK) and OrganiCity to highlight the challenges of implementing APIs that serve cities as elements of civic infrastructures. The conceptualization of City APIs in this article deliberately draws on the fields of software studies, design theory, and smart city implementation to problematize City APIs as political, social, and technological interfaces at the same time. The aim is to conceptualize City APIs as interfaces that negotiate and reveal conflicting interests and aspirations within a city but that are neither conceptually nor technically limited to sustaining only one vision of how a city can be perceived or experienced.

Proprietary social media APIs and the architecture of the web

The function of APIs in the current architecture of the web (as the most public part of the Internet) has attracted wide attention from social and humanities scholars, especially focusing on social media APIs such as those of Facebook or Twitter. But social media APIs represent just one domain of application. More generally, APIs are standardized interfaces for the exchange of data that implement a particular grammar for talking to machines. They can facilitate machine-to-machine communication or enable human-to-machine communication. In the latter case, human intentions (gestures, queries, inputs) are represented in a machine-readable format, e.g. by "pushing a button on a screen" through a graphical user interface (GUI) commands on data are executed. Following the typology of interfaces in computing by Cramer and Fuller (2008: 149), APIs "determine relations between software and software," and are thus not necessarily discernible on the level of the GUI. They are designed to access specific functionalities and data points in simple or complex data types (e.g., numeric, boolean, string, position, address) and employ only a limited set of operations such as GET, PUT, DELETE, or POST, when using the Hypertext Transfer Protocol.

One consequence of network-based APIs is that they create a specific architectural style to connect

distributed resources and hosts for any given purpose or computation task in stable ways. APIs make "dataset[s] understandable by applications or websites" (Forum Virium, 2017: 9) and further implement the "weaving of the web" from dynamic content that was originally envisioned by Tim Berners-Lee for HTML encoding of websites (Berners-Lee and Fischetti, 2000). The most prevalent architectural style for web-based applications is "representational state transfer" (REST), originally defined by Roy Fielding. The REST protocol separates clients from servers, which means that development on each side can be done separately. The protocol is also stateless, which means each "call" contains all information necessary for its completion and therefore does not need status information from previous calls to be executed. Fielding associates the increased popularity of network-based APIs to "social dynamics" among developer communities in the late 1990s and a diversification of use cases for networked data (138f). In the so-called RESTful architecture of the web, APIs became essential components which allowed to dynamically "pull" data from different databases in response to user inputs or requests, instead of "pushing" the identical static information to every visitor of a page. What made REST popular was its flexibility and tolerance for third-party developers to build web applications and their mash-ups (the so-called programmable web, or Web 2.0; see Helmond, 2015). APIs have fundamentally contributed to what Plantin et al. (2016) have pointedly called a "'platformization' of infrastructures and an 'infrastructuralization' of platforms" (298) by standardizing exchanges for dynamic data between computer systems, platforms, and users (human and nonhuman). This standardization of data exchanges, however, has powerful consequences for the variety of applications that can be built when particular APIs are defined on a proprietary basis by only a few corporations, such as Facebook, Google, or Twitter, and begin to assume infrastructural functions for a broad range of applications. This development from data standardization through APIs to effective monopolization of access to data has been observed and criticized most fervently in the domain of social media research.

For researchers interested in tapping into the fabric of social life online (Borra and Rieder, 2014; boyd and Ellison, 2007; Manovich, 2011), APIs serve as "invisible actors" (Lahey, 2016) and central gatekeepers to what kind of data is publicly available. Because proprietary APIs are designed for commercial exploitation by third-party developers, control over API functions and access rights is with the corporations who define them. Appropriating such APIs for research questions creates a situation where a "data set may have many millions of pieces of data, but this does not mean it is random or representative" (boyd and Crawford, 2012: 668). Because the baseline or unit of comparison cannot be established independently of a proprietary social media API, big data research on social media sites requires considerable analytic and technical investment, skill development, and critical interrogation of automatically retrieved data (Kitchin, 2014a; Lomborg and Bechmann, 2014; Rieder and Gerlitz, 2013). As Bruns (2013) remarks in relation to Twitter, a vast collection of tweets, identified through a hashtag and retrieved through the Twitter Streaming API, may yield an "apparently well–delimited object of research which may not have been experienced in this form by any actual user" (see also González-Bailón et al., 2014). The problem is that a data collection on specific issues or relations of users will include only those items that match a certain search pattern and that *can* actually be provided by an API in the first place.

Criticism of proprietary social media APIs has focused mainly on issues of reliability and reproducibility of data collections, linking the design of the API to how a platform shapes access to the Web, gathers and processes user data, and privileges certain business models to be built on top of its data. APIs execute crucial gatekeeping functions which evade public scrutiny. Taking up the example of *Coord* again, specific APIs are designed by Sidewalk Labs to enable third-party applications to use only *Coord* data and functions, e.g. through the *Routing API* for calculating optimized journeys in real time between places or the *Tolls API* to factor in toll fares on journeys.[2] Because such APIs provide indispensable data for serving the visible layer of the user interface, they begin to assume infrastructural functions for navigating urban spaces although their governance and design is not subject to public scrutiny or even awareness. What kinds of factors are involved in the calculation of an optimal journey remains a business secret and intellectual property of *Coord*, although routing users in particular ways (via applications on mobile phones, for example) makes a difference in how certain urban spaces are perceived, are made (in)visible, or are accessed.

APIs then appear as "powerful governing techniques of the current social Web" (Bucher, 2013; see also Sandvig, 2013) that initially enabled the transition from networks of static web pages to platform-based architectures in the early 2000s (Helmond, 2015). Starting with Facebook's Developer API in 2006, the API principle enabled third-party applications to be "seamlessly integrated" with social media platforms, shaping the "industry-wide practice of *controlled openness and interoperability* among social media services" (Bodle, 2011: 329, emphasis added). Controlling access to a platform's data while building markets for such data makes the API function as a "gateway,

permitting other systems to interact with Facebook to form a seamlessly interactive network" (Plantin et al., 2016: 303). Through interactive elements such as the "Like" button for Facebook, an API-based infrastructure was implemented to analyze user preferences and behavior beyond the platform itself (Gerlitz and Helmond, 2013). Specialized APIs generate data points from all over the web to automate the creation of fine-grained user profiles that are made accessible for third-party developers on a contractual and proprietary basis, e.g. to enable the dissemination of advertising (Bechmann, 2013). Access to these APIs is governed by either keeping the documentation of API specifications from public view, by limiting access quota to data streams, or by enforcing the use of specific developer environments for which authentication is required.

An API functions as an element of infrastructure that represents a specific kind of management style, or what Galloway (2004) considers as "protocol": A set of commands regulating "autonomous agents who operat[e] according to certain pre-agreed 'scientific' rules of the system" (38). The API is a "protocological object" (Bucher, 2013) that standardizes and regulates data flows. Control over this object rests with the owners and designers of APIs, those who have to negotiate what types of data and functions are accessible, how they are described, and who can use them. Design decisions on data types, methods, and access quota define APIs as "anticipatory in their very operational logic" (Bucher, 2013). Just like the inventors that Hughes described as weaving a "seamless web" between science and business in the electricity sector around the turn of the 20th century, designers of APIs "attempt to anticipate" future uses for data and functions, while they need to "incorporate responses" for users of APIs that they will rarely encounter in person (Hughes, 1986: 290). Designing an API must therefore integrate technological requirements with anticipations on the user's side about how the underlying system operates, what resources it offers, and how these can be queried. Ash (2015: 28) has previously brought attention to the "process by which objects in interfaces are organized by designers to produce particular qualities for other objects in that interface and for the people using that interface". As Ash et al. (2018: 177) describe it, elements and tones of interfaces "are actively designed to work together to shape or prime specific responses or actions to potentially occur".

API developers have a central role in designing these interfaces and how they allow to access data, functions, or resources. In the context of a city or urban space, designing an API outside of a commercial or proprietary context will necessarily involve taking into account what kinds of data sources are available and who is likely to use them. Because an API operates at the level of defining and providing data access that serves as a prerequisite and condition for user-focused applications, its definition and implementation embeds crucial socio-political assumptions in a technological framework that has far-reaching consequences for citizens, city administrators, and developers of applications using social urban data. In the next part, we will therefore discuss in more detail the design process of an API through core concepts from human–computer interaction (HCI) design.

Designing APIs: Affordances and mental models of API producers and consumers

The process of designing an API needs to effectively communicate between available data assets, functionalities, and potential uses of an API. We here focus on the differences between *API producers*, i.e. developers of APIs, and *API consumers*, i.e. third-party developers who build applications that access data through APIs (De Souza et al., 2004). It is important to clarify that we do not discuss the role of users, at least not directly, in the process of designing APIs. A citizen using mobility data or seeking information on current events will likely access such information through an app or a web portal—an application designed for this specific purpose using API input or even nonstandardized data (e.g., user comments, manual updates). Users are thus only indirectly involved in the design and offerings of an API through their very behavior within an app or a portal and the analysis of their behavior through an API consumer, i.e. the developer of an app or portal. The design process which we highlight here concentrates on the terms in which the API frames and defines concepts such as "mobility" or "current event," which is a negotiation between API producers and API consumers—not users or citizens directly. In this perspective the API then appears as an element of an urban information infrastructure that is less a fixed object of centralized control but more a momentarily stabilized convention to support the seamless integration of various data sources for specific purposes.

The convention established between API producers and consumers can be framed in design terms as the *affordance* of an API. Understanding APIs as interfaces between humans and computers emerged in the early 2000s and has since evolved in academic as well as practical fields of API design (Myers and Stylos, 2016).[3] Terms like *API usability* and *DevX* (as an analogy to user experience or *UX*) underline the importance of usability in API development. API design has many common concerns around affordances with the interaction design field, whose methods and approaches

are embraced by API communities, e.g. the Nordic APIs community with a broad portfolio of open access publications on API design approaches.[4]

Initially, the term affordance was coined by the psychologist James Jerome Gibson (1979) to describe in phenomenological terms how animals perceive objects differently depending on contextual affordances, insinuating that the affordance of an object such as a log wood could be directly perceived (127–128). The design theorist Donald Norman (1988, 1998) adopted Gibson's term and introduced it to the field of design. He initially defined affordances as "the perceived and actual properties of the thing, primarily those fundamental properties that determine just how the thing could possibly be used" (Norman, 1988: 9). This definition suggests that affordances are identical to properties of objects. However, as Norman (2013) specified in the revised edition of *The Design of Everyday Things* (2013), affordances describe "a relationship between a physical object and a person", which is "*jointly determined by the qualities of the object and the abilities of the agent that is interacting*" (11, emphasis added).

For API producers, an API needs to encode rules of how particular data objects are described and can be accessed, while anticipating different abilities of users to interpret these rules. In other words, an API's affordance is determined by how it translates real-world concepts such as traffic flow or occupancy of public buildings to data objects that are meaningful to different kinds of users, foregrounding the process of "encapsulating meaning" in an API itself (Holmboe, 2005: 189). One example for this process of encapsulating meaning can be the "Transport API,"[5] developed by Future Cities Catapult in London, which offers a range of data sets and real-time data sources from across the UK to potential API consumers but also citizens and public administrators. Available as an open data resource, the API allows to develop varied scenarios for the commercial and noncommercial use of such data, aligning restrictions on usage quota with levels of professionalization and commercialization on the side of API consumers. The example illustrates how an API can encapsulate different meanings of "transport" and can serve different affordances depending on the context and intended use case.

In Gibson's view, an object only had static material qualities that remained the same throughout its life. For APIs, the material qualities themselves are part of the environment and context in which they are perceived. In the words of design theorists Löwgren and Stolterman, APIs are "a material without qualities" since these qualities are constantly changing, either by altering existing functionalities, by deleting deprecated ones or adding new features (Löwgren and Stolterman,

2007: 3–4). This flexibility demands technological expertise as well as high context awareness for API producers and API consumers alike. What appears as a source of great flexibility to experienced developers can be daunting for novices like self-taught makers (Gauntlett and Holroyd, 2014), creative coders (Maeda, 2004), and interaction designers trying to bring APIs under their control (Schaeffer and Lindell, 2015: 718). Experienced API producers will refer to API documentations mainly to grasp its main concepts while untrained API consumers will need more experience with actual coding to manage the API interaction process successfully (Meng et al., 2017; Piccioni et al., 2013: 11). When an API encapsulates an ambivalent meaning or inadequately describes a real-world concept, interaction with an API can become highly unpredictable, e.g. when a current sensor reading is expected by an API consumer but the API returns only historical data or no data at all because the query is wrongly expressed. The specification of an API can also change over time, altering its affordance and thereby changing the use case. API producers and consumers therefore need to negotiate and agree on the domain which the API encapsulates, e.g. which data objects, assets, or functionalities are made available based on which standards or metadata frameworks.

The concept of affordance as an active negotiation of properties of objects with user capabilities brings to the fore that perceptions of real-world phenomena such as traffic or air quality can be interpreted in quite different ways when translated into data terms or seams. Give students the open task of defining functionalities of a "Library API," for example, and they will come up with different ideas about what purpose this API shall serve: it may refer to either the media on the stacks, the building itself, or even properties of employees. A widely known real-world concept like "library" can be understood in very different ways when translated into data. The design of an API needs to address this and reveal which specific interpretation is used, or what is called in design terms the *conceptual model* of a library, of transport, or air quality. API design is therefore not simply a straightforward task to describe and make existing data objects accessible. Usability of an API needs to effectively address and anticipate the assumptions of varied user groups (see Bloch, 2006; Grill et al., 2012; Myers and Stylos, 2016; Scheller and Kühn, 2015) and reveal the intent of an API effectively, "[providing] information about why certain API design decisions were made, and how the API is intended to be used" (Robillard and DeLine, 2011: 716). The conceptual model and the intent of an API need to be sufficiently described by API producers, either in their choice of categories for the description of data or in the documentation, to

reduce the risk of inappropriate or insecure uses among API consumers.

Because API producers and consumers rarely interact directly, the design process of an API needs to reflect both the capabilities of a data source and the uses it can be put to. An API producer will first create an API specification (the conceptual model) and then implement the specification into a usable API (the system image). Whereas the conceptual model only outlines how different components of the system are connected and described, the working API needs to reveal a system image of what the underlying infrastructure is capable of (Balijepally et al., 2012: 5442). In this sense, the technical specifications of an API implement design choices about how the interface is appropriately accessed and what resources it makes available. There are data models available even for flowerbeds which could be embedded in an API, detailing in what ways such a quotidian, real-world object can be described in terms of data.[6] But using this specific data model will eventually capture only data that is expressible in terms of the model and not necessarily in terms of how a flowerbed is maintained, used, or even experienced by different people. Revealing intent effectively through an API depends on a high level of congruence between the conceptual model of API functionalities on the side of a producer, the system image that is created by the actual implementation of the API, and the anticipations about API functionalities by API consumers. These anticipations among users (here: consumers) are called "mental models" in design theory.

Building on the work of Johnson-Laird (1983) and others, the concept of mental models is central in design theory (Norman, 1988, 2013) and is meanwhile also used to study the dynamics of collaboration in distributed developer teams (Balijepally et al., 2012; DeFranco et al., 2011; Scozzi et al., 2008). In design theory "[t]he mental model of a device is formed largely by interpreting its perceived actions and its visible structure" (Norman, 1988: 17). A mental model of an action or an object is built from previous experiences with similar objects or actions and helps to understand new situations even if the specifics are different (Norman, 1988: 70–71). The design process of an API can be regarded as a negotiation of mental models between API producers and consumers. Whereas the former need to design APIs that reveal the appropriate system image, e.g. what data is available in which format about which flowerbeds, the latter build mental models of the system image by interacting with it (Norman, 1988: 16), e.g. by evaluating the quality and scope of data available about flowerbeds in a particular location. Mental models are important for understanding how design choices influence the usability of APIs. Since the API serves as the intermediate between a data source and its consumers, the specific functions, parameters, and return values of an API all affect what mental model API consumers will develop about the underlying system (Robillard and DeLine, 2011: 722–723). This puts a high responsibility on API producers to anticipate future API consumers' expectations and needs. As Norman (1988) states: "If the system image does not make the design [conceptual] model clear and consistent, then the user will end up with the wrong model" (16). Designers need to evoke and reveal a system image that adequately fulfills API consumers' mental model, i.e. fulfills their expectation of what kind of data can be obtained and how it can be queried through API calls.

The design of APIs is therefore a process of negotiating affordances, mental models, and system images to reveal a specific intent. This can happen through, e.g. appropriate documentation or facilitating the repeated interaction with APIs. While it is tempting to discuss guidelines for developing successful APIs[7] the aim of this article is to problematize how interfaces to urban information infrastructures such as City APIs are the results of negotiations between design options, political objectives, technological capabilities, and civic values. Deciding what affordances City APIs shall offer, what mental models they raise among its varied consumers, and what seams they reveal within the chaotic and contradictory urban sphere is a highly contested domain. Again, there is not one City API which is applicable for all cities or use cases but design choices about APIs can reveal different seams in every city, depending on the political, social, and cultural imperatives that shape the design specifications for APIs and the mental models involved in the process. City APIs can be seen as a specific technological manifestation of political and design choices that actively anticipate a particular course and potential use of social urban data for a broad range of stakeholders.

City APIs: From public services to civic infrastructures

Applying the concept of an API to the seams of data woven into the fabric of a city brings up the question what future purposes such an interface to data shall serve and for whom. In contrast to commercial or proprietary APIs, often developed by only one company, designing interfaces for data seams in cities needs to include broader deliberations across actor groups—from administrators and politicians to citizens and enterprises—to determine the objectives of making data available. Because interests and aspirations around the use and potentials of public data can vary greatly within a city, developing interfaces has to fulfill

competing demands for optimized public service delivery, innovation of businesses, as well as supporting accessibility for and participation by citizens. Although an API is often designed for a specific use case, deliberation on these competing demands and multiple uses in a city can serve also to identify current disjunctions between branches of city administration, aspirations for civic participation, and the needs of business and research. City APIs can thus be conceived as crucial elements in civic infrastructures that potentially also enable new ways making the city "hackable" (De Waal et al., 2017) and open to interventions by citizens (Mattern, 2014, 2017a). The main goal of this third part is to conceptualize City APIs as interfaces to urban information infrastructures that can support these varied (and often conflicting) public, civic, and commercial uses at the same time.

In popular and academic discussions, the idea of seamless integration of data sources through APIs is often associated with an imaginary of a city dashboard—a centralized command center for surveillance, monitoring, and management. As Kitchin et al. (2015) point out, dashboard imaginaries "privilege a realist, quantitative epistemology, enact an instrumental rationality, and fail to recognize and denote their contingent unfolding, their inherent politics, and their technical and methodological issues" (24). The dashboard as an imaginary, and meanwhile also *actual* feature of city governance (e.g., in Rio de Janeiro, London or Moscow), nurtures an understanding of City APIs as centralized interfaces to many different urban systems. However, this conception puts the cart before the horse. A dashboard is only *one* particular application built on top of various seams of data with their own API specifications, integrating, for example, data on weather conditions, road traffic, and occupancy of public spaces. The dashboard is a specific GUI for managing and monitoring urban processes in real time (see, e.g., Gray et al., 2016) with a dominant emphasis on visualizing complex sources of data to yield insights into spatio-temporal patterns and processes (Ferreira et al., 2013; Mattern, 2015). The dashboard makes information overload manageable and "allows the user to step into and out of a number of different streams in an intuitive and natural way" (Beer, 2011: 142). The design of a dashboard depends in large part on the anticipated uses of this interface, which necessarily differs between traffic management or experimental ways of enabling citizens to use their cell phone as an interface to the city (Usurelu and Pop, 2017). While a range of specific APIs can feed a centralized dashboard for monitoring and managing urban flows, the design of City APIs can conceptually also highlight which seams are woven from which kind of data, which sources were used, and which civic or

commercial uses were privileged. Instead of ignoring the social context in which data is collected and analyzed, an API and its applications could also include small barriers to make "people more mindful of their behavior" in balance with "facilitating efficient everyday interface interactions" (Cox et al., 2016: 1394). In contrast to a dashboard imaginary of centralized control, City APIs can then also be envisioned as interfaces that reveal the *diversity* of seams of data within a city as well as the social contexts in which particular insights, experiences, or interactions emerge.

The importance of APIs as integral infrastructural elements of web-based communication also signals a shift from the common practice of publishing static data sets in data repositories to integrating real-time, dynamic data such as traffic information, weather conditions, or occupancy of public spaces. Because development and design costs are much higher, publishing data through APIs implies the continuous update of such data whereas the publishing of static data sets as open government data (OGD) often merely reflects redundant data no longer in use in city departments. To give an example: There are currently more than 880,000 open data sets from cities all over Europe listed in the European Data Portal.[8] These data sets vary greatly in terms of geographical distribution, subject area, and degree of standardization. Despite the problems with such unstructured and often incomparable data, OGD is often seen as "facilitat[ing] new marketplaces for software entrepreneurship" (Barns, 2015: 556), to enable more efficient public services, promote accountable government action, and provide an open resource for citizens to participate in public deliberation (Aguilera et al., 2017; Ubaldi, 2013). The problem with sharing data sets in such a way is that they are mostly collected for specific departmental purposes and are not necessarily *designed* for subsequent usage. By making urban data available through APIs rather than as static data sets, the design of interfaces requires to some degree cross-departmental (and even city-wide) deliberation about future uses and resources which may be of interest for more than one group of stakeholders.

As pointed out by Kitchin (2014b), the technical proficiency to create and manage APIs for urban data, in contrast to publishing static data sets, can obstruct the availability of such data and its future uses outside of core public services. Take the example of the German town Potsdam (about 170,000 inhabitants). Challenged with the task of establishing an open data portal, the city initiated a public survey in 2015 to determine which kind of data would be interesting for citizens, researchers, and businesses. One result of this survey was that cartographic data for current and prospective developments was high in demand, with

preferences for pdf formats (likely to be demanded by citizens) and in standardized GIS data (likely to be demanded by developers and researchers). The city commissioned a service provider (OpenDataSoft) to set up and maintain the portal, which went online in 2018, three years after the survey.[9] Cartographic data is now available in several static formats as well as through an API, which allows subsequent public and commercial services to be built on top of this data. The example illustrates that the design of City APIs is dependent on a much broader internal and public deliberation process through new kinds of collaborative and participatory practices, in which divergent aspirations and affordances among stakeholders in a city can be articulated and negotiated. The example also shows that the process of establishing public and cross-departmental collaborations around a vital subject like open data can take quite a long time to materialize in a usable interface, often facilitated through newly established departments or units.

To illustrate the role of City APIs as elements in civic infrastructures we discuss two broader initiatives in the domain of smart city innovation: CitySDK and OrganiCity. In both initiatives, APIs are central to a design process that foregrounds the future uses of urban data *across cities*, creating a public and civic resource outside of the departmental domain of each city government but also enabling a city-wide market for specialized applications based on urban data.

The CitySDK[10] was developed by a consortium of eight European cities to integrate the development of APIs for common tasks in cities. Led by Forum Virium in Helsinki, CitySDK published the ''Harmonised Smart City API's'' for common tasks such as issue reporting by citizens, event announcements, linked data for transportation and mobility, or data for tourists. Because APIs were designed for tasks that are similar in many cities, they reduce the effort needed to create such services individually in each city while setting interoperability standards for third-party services using these APIs (Forum Virium, 2017). Rather than starting from the availability of existing data sets within one city (e.g., through OGD in data repositories), the design of these City APIs was geared toward portability and interoperability between cities for common challenges and use cases. This example demonstrates how the initial task of optimizing a public service (e.g., issue reporting) helped to initiate an infrastructural development effort across different cities to avoid the lock-ins of proprietary APIs and to align the design of applications with cities' needs and aspirations.

In the second project (OrganiCity), the aim was to establish an Experimentation-as-a-Service facility for public and sensor data across the cities of Aarhus, London, and Santander (see Brynskov et al., 2018; Gutierrez et al. 2017). The project was based on specific challenges posed by the participating cities and developed a joint platform and tools for sharing data. Through two open calls the project enabled teams of experimenters to experiment with new scenarios for such data through co-creation methodologies and a dedicated community-outreach campaign geared toward nonexperts and the general public. Participants in the project and the experiment teams followed a set of engagement principles to enable community involvement, foster network building, and collect local knowledge and experiences. The OrganiCity platform was developed to enable very different kinds of expert and nonexpert users of data to experiment with and prototype new use cases and scenarios that addressed city challenges. One outcome of the co-creation processes was the development of APIs to discover and annotate data sources in cities. The Asset Discovery API[11] defines access to various static and real-time data sources from all participating cities, enabling experimenters to compare and integrate data sources for new uses. The Asset Annotation API[12] allows the description of data assets in natural language tags, similar to tagging photos or posts on social media, creating a crowdsourced resource where nonexpert users can express multiple identities of data points, highlighting their respective contexts and possible fields of application.

CitySDK and OrganiCity show that publishing urban data through APIs initiates a design and development process for particular challenges or objectives that can be co-created by citizens, city officials, researchers, and businesses rather than being the sole development effort of large corporations (cf. Brynskov et al., 2018; Gutierrez et al., 2017). Based on standardized descriptions of data and interfaces across cities, and by integrating common methods to combine data sources for specific tasks, APIs can serve both improved public service delivery as well as enabling new civic and commercial uses. In the city of Aarhus, to give another example, 1000 district heating wells were equipped with low-cost sensors to monitor leaks and temperature losses within the system. The sensors and gateways use open LoRaWan standards and specifications (Low Power Wide Area Network Technology)[13] and are estimated to save Aarhus municipality about 1 million Danish Kroner annually (US$152,000 or £119,000).[14] Where an experimental approach to solving such urban challenges can be fostered through standardized interfaces and data, a viable and working solution can quickly develop into best practices and potentially new markets across cities. But developing such standards for APIs, data models, and management processes is often beyond the scope of individual

projects or cities. The example of Potsdam shows that domain expertise in open data portals and API specifications is not readily available within a city and needs to be acquired from external service providers. The example of Aarhus, in contrast, demonstrates that a city can decide to build its own specialized, cross-departmental unit and build up internal competence in this field.

It is even more difficult to integrate the plurality of experimental and local solutions in broader transnational or cross-domain initiatives and standardization bodies, e.g. in the Smart Cities and Community Study Group under the International Telecommunication Union, UN initiatives such as United 4 Smart Sustainable Cities, the European Telecommunications Standards Institute, or the International Organization for Standardization. Associations of cities like Open & Agile Smart Cities seek to scale solutions and developments in projects such as CitySDK or OrganiCity, to enable the reuse of APIs and data models to develop minimal interoperability mechanisms to create inter-city markets, exchanges, and infrastructures.[15]

City APIs are vital elements of urban information infrastructures in which the city appears increasingly as an "ecosystem of interconnected digital media platforms" (De Waal et al., 2017: 52) that do not necessarily overlap with physical locations or their inhabitants. To maintain a sense of connection between these platforms and their users, between digital domain spaces and physical locations, a stronger attention to the politics and designs of interfaces that negotiate these connections is needed. Urban theorists and activists have challenged the smart city paradigm precisely for its technocratic, instrumentalist vision, and argued fervently for an appreciation and protection of the plurality of urban life. The question is here, how can new urban information infrastructures, deployed at scale and at immense public cost, be designed so as to offer a "peek down the urban stack" (Mattern, 2014) and to allow interventionist approaches to "city making" and "city hacking" (Brynskov et al., 2014; De Waal, 2014; De Waal et al., 2017; Estrada-Grajales et al., 2018; Foth et al., 2016; Mattern, 2017a, 2017b; Van de Moere and Hill, 2012). In light of large-scale deployments of sensor infrastructures such as Sidewalk Labs' planned and contested redevelopment of the Toronto waterfront,[16] there is a growing perception that "the rhetoric and aspiration of smart cities attempts to obviate civic engagement because the political axes of history, justice, and culture are rendered illegible through instrumentation" (Asad et al., 2017: 2302). In this article we have argued that City APIs can include more than one vision of what seams get woven from social urban data to make cities meaningful from more than

one, efficiency-driven angle and interpretation of seamlessness. We have also argued that commercial applications do not need to contradict civic aspirations around these infrastructures. Designing CityAPIs that fulfill conflicting commercial and civic demands will necessarily reveal conflicts between different groups of stakeholders in a city. What we have underlined in this article is that the actual implementation of CityAPIs for specific services and their conceptual implications as elements of civic infrastructures cannot be dissociated from questions of design, governance, and access.

Summary

We have offered three analytic inroads to conceptualizing APIs as central elements of infrastructures. In an applied sense, City APIs describe actual implementations of interfaces for social urban data that enable a range of inter-city services and standardizing particular routines of access while remaining open to reuse and appropriation by users. In which ways an API creates mental images and affordances for its use was discussed in the second part, which highlighted that the development of APIs was a specific design challenge in which system capabilities, user anticipation, and documentation reflexively shape APIs as "protocological objects" (Bucher, 2013). Through this perspective of design we have also emphasized that such objects appear as less fixed and easy to identify when seen from the processual logic of their development. What appears as more or less stable from the point of critical analysis (e.g., proprietary APIs) is from a design and implementation perspective revealed as a momentary stabilization of conventions in an ongoing process of negotiation. Which affordances are offered through an API is thus a direct result of the kinds of seams that are selected for its operation and its anticipated future uses. Understanding APIs as reflexive objects that negotiate between users and systems, we can see how our knowledge of such objects is permeated by a shift from *mimetic* to *poietic* ways of knowing. It has shifted from a "description of reality" through analytic science to a "conceptual logic of construction that does not start from the system to analyse ... in terms of a model, but actually starts from the model (the blueprint) to realise the system." Designing then is significant as a way of knowing, of speculation and implementation, it becomes "an independent epistemic praxis through which one can acquire genuine *ab anteriori* knowledge" (Floridi, 2017: 508). Acknowledging the complementarity of this epistemic praxis with different domains of scholarship in big data research allows to conceptualize City APIs in relation to technological, political, and social contestations over which

future uses and modes of access to civic infrastructures are embedded in today's designs of interfaces.

Notes

1. https://www.sidewalklabs.com/blog/announcing-coord-the-integration-platform-for-mobility-providers-navigation-tools-and-urban-infrastructure/.
2. https://coord.co/showcase.
3. http://apiusability.org was established in 2009, and holds a rich body of academic references to API usability.
4. https://nordicapis.com/designing-apis-humans/.
5. https://www.transportapi.com/.
6. https://fiware-datamodels.readthedocs.io/en/latest/ParksAndGardens/FlowerBed/doc/spec/index.html.
7. https://sites.google.com/site/apiusability/resources offers an overview of API development from a design perspective.
8. https://www.europeandataportal.eu/data/dataset (accessed 30 December 2018). For an assessment of the standardization in public data sets in the US, see Barbosa et al. (2014).
9. https://www.potsdam.de/kategorie/open-data.
10. https://www.citysdk.eu/about-the-project/.
11. https://docs.organicity.eu/api/AssetDiscovery.html.
12. https://annotations.organicity.eu/swagger-ui.html.
13. See https://www.thethingsnetwork.org/.
14. https://www.smartaarhus.eu/projects/narrowband-network-aarhus.
15. http://oascities.org/about-oasc/.
16. https://www.curbed.com/2018/11/30/18119679/toronto-sidewalk-labs-toronto-waterfront-alphabet.

References

Aguilera U, Peña O, Belmonte O, et al. (2017) Citizen-centric data services for smarter cities. *Future Generation Computer Systems* 76: 234–247.

Asad M, Dantec CAL, Nielsen B, et al. (2017) Creating a sociotechnical API: Designing city-scale community engagement. In: *Proceedings from CHI*, 6–11 May 2017.

Ash J (2015) *The Interface Envelope: Gaming, Technology, Power*. New York: Bloomsbury Academic.

Ash J, Anderson B, Gordon R, et al. (2018) Unit, vibration, tone: A post-phenomenological method for researching digital interfaces. *Cultural Geographies* 25(1): 165–181.

Balijepally V, Nerur S and Mahapatra R (2012) Effect of task mental models on software developer's performance: An experimental investigation. In: *The 45th Hawaii international conference on system sciences*, Maui, pp. 5442–5451. DOI: 10.1109/HICSS.2012.9.

Barbosa L, Pham K, Silva C, et al. (2014) Structured open urban data: Understanding the landscape. *Big Data* 2(3): 144–154.

Barns S (2015) Mine your data: Open data, digital strategies and entrepreneurial governance by code. *Urban Geography* 37(4): 554–571.

Batty M (2013) *The New Science of Cities*. Cambridge: MIT Press.

Bechmann A (2013) Internet profiling: The economy of data intraoperability on facebook and google. *MedieKultur* 29(55): 72–91.

Beer D (2011) *The Philosophy of Software. Code and Mediation in the Digital Age*. Basingstoke: Palgrave Macmillan.

Berners-Lee T and Fischetti M (2000) *Weaving the Web. The Original Design and Ultimate Destiny of the World Wide Web by Its Inventor*. San Francisco, CA: HarperBusiness.

Bloch J (2006) How to design a good API and why it matters. In: *Companion to the 21st ACM SIGPLAN conference on object-oriented programming systems, languages, and applications (OOPSLA '06)*, New York, pp. 506–507. Portland, Oregon: ACM Press. DOI: 10.1145/1176617.1176622.

Bodle R (2011) Regimes of sharing: Open APIs, interoperability, and facebook. *Information, Communication & Society* 14(3): 320–337.

Borra E and Rieder B (2014) Programmed method: Developing a toolset for capturing and analyzing tweets. *ASLIB Journal of Information Management* 66(3): 262–278.

Bowker GC, Baker K, Millerand F, et al. (2010) Toward information infrastructure studies: Ways of knowing in a networked environment. In: Hunsinger J, Klastrup L and Allen M (eds) *International Handbook of Internet Research*. Dordrecht: Springer, pp. 97–117.

boyd DM and Crawford K (2012) Critical questions for big data: Provocations for a cultural, technological, and scholarly phenomenon. *Information, Communication & Society* 15(5): 662–679.

boyd DM and Ellison NB (2007) Social network sites: Definition, history, and scholarship. *Journal of Computer-Mediated Communication* 13(1): 210–230.

Bratton BH (2016) *The Stack: On Software and Sovereignty*. Cambridge: MIT Press.

Bruns A (2013) Faster than the speed of print: Reconciling "big data" social media analysis and academic scholarship. *First Monday* 18(10). Available at: http://firstmonday.org/ojs/index.php/fm/article/view/4879/3756 (accessed 23 January 2019).

Brynskov M, Bermúdez JCC, de Waal M, et al. (2014) *Urban Interaction Design. Towards City Making*. Amsterdam: Stichting Floss Manuals. Available at: http://booksprints-for-ict-research.eu/wp-content/uploads/2014/06/Urban_Interaction_Design_Towards_City_Making.pdf (accessed 23 January 2019).

Brynskov M, Heijnen A, Balestrini M and Raetzsch C (2018) Experimentation at scale: Challenges for making urban informatics work. *Smart and Sustainable Built Environment* 7(1): 150–163. DOI: 10.1108/SASBE-10-2017-0054.

Bucher T (2013) Objects of intense feeling: The case of the twitter API. *Computational Culture* 3. Available at: http://computationalculture.net/objects-of-intense-feeling-the-case-of-the-twitter-api/ (accessed 23 January 2019).

Coletta C and Kitchin R (2017) Algorhythmic governance: Regulating the 'heartbeat' of a city using the internet of things. *Big Data & Society* 4(2): 1–16. DOI: 10.1177/2053951717742418.

Cox AL, Gould SJJ, Cecchinato ME, et al. (2016) Design frictions for mindful interactions. In: *Proceedings of the 2016 CHI conference, extended abstracts on human factors in computing systems*, San Jose, 7–12 May.

Cramer F and Fuller M (2008) "Interface". In: Fuller M (ed.) *Software Studies: A Lexicon*. Cambridge: The MIT Press, pp. 149–153.

De Souza CRB, Redmiles D, Cheng L-T, et al. (2004) Sometimes you need to see through walls: A field study of application programming interfaces. In: *Proceedings from CSCW '04 ACM conference on computer supported cooperative work*, New York, 8–10 November.

De Waal M (2014) *The City as Interface: How New Media are Changing the City*. Rotterdam: NAI Publishers.

De Waal M, de Lange M and Bouw M (2017) The hackable city: Citymaking in a platform society. *Architectural Design* 87(1): 50–57.

Estrada-Grajales C, Foth M and Mitchell P (2018) Urban imaginaries of co-creating the city: Local activism meets citizen peer-production. *Journal of Peer Production* 11. Available at: http://peerproduction.net/issues/issue-11-city/peer-reviewed-papers/urban-imaginaries-of-co-creating-the-city/ (accessed 23 January 2019).

Ferreira N, Poco J, Vo HT, et al. (2013) Visual exploration of big spatio-temporal urban data: A study of New York City taxi trips. *IEEE Transactions on Visualization and Computer Graphics* 19(12): 2149–2158.

Fielding RT (2000) *Architectural styles and the design of network-based software architectures*. Doctoral dissertation, University of California, USA. Available at: https://www.ics.uci.edu/~fielding/pubs/dissertation/fielding_dissertation.pdf (accessed 23 January 2019).

Floridi L (2017) The logic of design as a conceptual logic of information. *Minds & Machines* 27(3): 495–519.

Forum Virium (2017) *Harmonised Smart City Apis: A Cook Book for Cities*. Helsinki: Forum Virium. Available at: https://www.citysdk.eu/wp-content/uploads/2016/12/20171109_HarmonisedSmartCityAPIs_WEB.pdf (accessed 23 January 2019).

Foth M, Brynskov M and Ojala T (eds) (2015) *Citizen's Right to the Digital City: Urban Interfaces, Activism, and Placemaking*. Wiesbaden: Springer.

Foth M, Forlano L, Satchell C, et al. (eds) (2011) *From Social Butterfly to Engaged Citizen: Urban Informatics, Social Media, Ubiquitous Computing, and Mobile Technology to Support Citizen Engagement*. Cambridge: MIT Press.

Foth M, Hudson-Smith A and Gifford D (2016) Smart cities, social capital, and citizens at play: A critique and a way forward. In: Olleros F and Zhegu M (eds) *Research Handbook on Digital Transformations*. Cheltenham: Edward Elgar Publishing, pp. 203–222.

Foth M, Rittenbruch M, Robinson R, et al. (eds) (2014) *Street Computing: Urban Informatics and City Interfaces*. London and New York: Routledge.

DeFranco JF, Neill CJ and Clariana RB (2011) A cognitive collaborative model to improve performance in engineering teams: A study of team outcomes and mental model sharing. *Systems Engineering* 14(3): 267–278.

Galloway AR (2004) *Protocol: How Control Exists After Decentralization*. Cambridge: MIT Press.

Gauntlett D and Holroyd AT (2014) On making, sustainability and the importance of small steps: A conversation. *Conjunctions* 1(1): 1–27.

Gerlitz C and Helmond A (2013) The like economy: Social buttons and the data-intensive web. *New Media & Society* 15(8): 1348–1365.

Gibson JJ (1979) *The Ecological Approach to Visual Perception*. Boston, MA: Houghton Mifflin.

González-Bailón S, Wang N, Rivero A, et al. (2014) Assessing the bias in samples of large online networks. *Social Networks* 38: 16–27.

Graham SDN (2016) Software-sorted geographies. *Progress in Human Geography* 29(5): 562–580.

Gray S, O'Brien O and Hügel S (2016) Collecting and visualizing real-time urban data through city dashboards. *Built Environment* 42(3): 498–509.

Grill T, Polacek O and Tscheligi M (2012) Methods towards API usability: A structural analysis of usability problem categories. In: Hutchison D, et al. (eds) *Lecture Notes in Computer Science*. Heidelberg: Springer, pp. 164–180.

Gutierrez V, Amaxilatis D, Mylonas G, et al. (2017) Empowering citizens towards the co-creation of sustainable cities. *IEEE Internet Things Journal* 5(2): 668–676. DOI: 10.1109/JIOT.2017.2743783.

Helmond A (2015) The platformization of the web: Making web data platform ready. *Social Media + Society* 1(2). DOI: 10.1177/2056305115603080.

Holmboe C (2005) The linguistics of object-oriented design. In: *Proceedings from ITiCSE'05*, New York, 27–29 June,

Caparica, Portugal. New York: ACM, pp.188–192. DOI: 10.1145/1067445.1067498.

Hookway B (2014) *Interface*. Cambridge: The MIT Press.

Hughes TP (1986) The seamless web: Technology, science, etcetera, etcetera. *Social Studies of Science* 16(2): 281–292.

Johnson-Laird P (1983) *Mental Models. Towards a Cognitive Science of Language, Inference and Consciousness*. Cambridge, MA: Harvard University Press.

Kitchin R (2014a) *The Data Revolution: Big Data, Open Data, Data Infrastructures and Their Consequences*. Los Angeles, CA: Sage.

Kitchin R (2014b) The real-time city? Big data and smart urbanism. *GeoJournal* 79(1): 1–14.

Kitchin R (2018) The realtimeness of smart cities. *Tecnoscienza* 8(2): 19–41.

Kitchin R, Lauriault TP and McArdle G (2015) Knowing and governing cities through urban indicators, city benchmarking and real-time dashboards. *Regional Studies, Regional Science* 2(1): 6–28.

Kitchin R, Lauriault TP and McArdle G (eds) (2017) *Data and the City*. Abingdon: Routledge.

Kitchin R and Perng S-Y (eds) (2016) *Code and the City*. Abingdon: Routldege.

Lahey M (2016) Invisible actors: Web application programming interfaces, television, and social media. *Convergence* 22(4): 426–439.

Lomborg S and Bechmann A (2014) Using APIs for data collection on social media. *The Information Society* 30(4): 256–265.

Löwgren J and Stolterman E (2007) *Thoughtful Interaction Design: A Design Perspective on Information Technology*. Cambridge: MIT Press.

Maeda J (2004) *Creative Code: Aesthetics + Computation*. London: Thames & Hudson.

Manovich L (2011) Trending: The promises and the challenges of big social data. Available at: http://manovich. net/content/04-projects/067-trending-the-promises-and-the-challenges-of-big-social-data/64-article-2011.pdf (accessed 23 January 2019).

Mattern S (2014) Interfacing urban intelligence. *Places Journal*, April. Available at: https://placesjournal.org/article/interfacing-urban-intelligence/ (accessed 23 January 2019).

Mattern S (2015) Mission control: A history of the urban dashboard. *Places Journal*, March. Available at: https:// placesjournal.org/article/mission-control-a-history-of-the-urban-dashboard/ (accessed 23 January 2019).

Mattern S (2017a) A city is not a computer. *Places Journal* February. Available at: https://placesjournal.org/article/a-city-is-not-a-computer/ (accessed 23 January 2019).

Mattern S (2017b) *Code and Clay, Data and Dirt: Five Thousand Years of Urban Media*. Minneapolis: University of Minnesota Press.

Meng M, Steinhardt S and Schubert A (2017) Application programming interface documentation: What do software developers want. *Journal of Technical Writing and Communication* 48(3): 295–330.

Myers BA and Stylos J (2016) Improving API usability. *Communications ACM* 59(6): 62–69.

Norman DA (1988) *The Design of Everyday Things*, 1st ed. New York: Doubleday.

Norman DA (1998) *The Invisible Computer: Why Good Products Can Fail, the Personal Computer Is so Complex, and Information Appliances Are the Solution*. Cambridge: MIT Press.

Norman DA (2013) *The Design of Everyday Things*. Revised and expanded ed. New York: Basic Books.

Piccioni M, Furia CA and Meyer B (2013) An empirical study of API usability. In: *Proceedings from empirical software engineering and measurement*, Baltimore, 10–11 October.

Plantin J-C, Lagoze C, Edwards PN, et al. (2016) Infrastructure studies meet platform studies in the age of Google and Facebook. *New Media & Society* 20(1): 293–310.

Psyllidis A (2016) Revisiting urban dynamics through social urban data. Methods and tools for data integration, visualization, and exploratory analysis to understand the spatiotemporal dynamics of human activity in cities. *A+BE: Architecture and the Built Environment* 6(18): 1–334.

Rieder B and Gerlitz C (2013) Mining one percent of twitter: Collections, baselines, sampling. *M/C Journal* 16(2). Available at: http://journal.media-culture.org.au/ index.php/mcjournal/article/viewArticle/620 (accessed 23 January 2019).

Robillard MP and DeLine R (2011) A field study of API learning obstacles. *Empirical Software Engineering* 16(6): 703–732.

Robinson R, Rittenbruch M, Foth M, et al. (2012) Street computing: Towards an integrated open data application programming interface (API) for cities. *Journal of Urban Technology* 19(2): 1–23.

Sandvig C (2013) The internet as infrastructure. In: Dutton WH (ed.) *The Oxford Handbook of Internet Studies*. New York: Oxford University Press, pp. 86–106.

Schaeffer J and Lindell R (2015) Arduino in museum exhibition. In: *Proceedings from ninth international conference on tangible, embedded, and embodied interaction – TEI '14*, New York.

Scheller T and Kühn E (2015) Automated measurement of API usability: The API concepts framework. *Information and Software Technology* 61: 145–162.

Scozzi B, Crowston K, Eseryel UY, et al. (2008) Shared mental models among open source software developers. In: *Proceedings of the 41st annual Hawaii international conference on system sciences*, HICSS, 2008.

Sidewalk Labs (2018) Announcing COORD: The integration platform for mobility providers, navigation tools, and urban infrastructure. *Sidewalk Labs Blog*, 1 February 2018. Available at: https://www.sidewalklabs.com/blog/ announcing-coord-the-integration-platform-for-mobility-providers-navigation-tools-and-urban-infrastructure/ (accessed 5 March 2018).

Star SL (1999) The ethnography of infrastructure. *American Behavioral Scientist* 43(3): 377–391.

Star SL (2002) Infrastructure and ethnographic practice: Working on the fringes. *Scandinavian Journal of Information Systems* 14(2): 107–122.

Star SL and Ruhleder K (1996) Steps toward an ecology of infrastructure: Design and access for large information spaces. *Information Systems Research* 7(1): 111–134.

Sudjic D (2016) *The Language of Cities*. London: Penguin UK.

Ubaldi B (2013) *Open Government Data: Towards Empirical Analysis of Open Government Data Initiatives*. OECD Working Papers on Public Governance. Paris: OECD Publishing. DOI: 10.1787/5k46bj4f03s7-en.

Usurelu CC and Pop F (2017) My city dashboard: Real-time data processing platform for smart cities. *Journal of Telecommunications and Information Technology* 2017(1): 89–100.

Van de Moere A and Hill D (2012) Designing for the situated and public visualization of urban data. *Journal of Urban Technology* 19(2): 25–46.

Van Es K and Schäfer MT (eds) (2017) *The Datafied Society. Studying Culture through Data*. Amsterdam: Amsterdam University Press.

Wadmann S and Hoeyer K (2018) Dangers of the digital fit: Rethinking seamlessness and social sustainability in data-intensive healthcare. *Big Data & Society* 5(1): 1–13.