

# Configuring the Independent Developer

---

05/07/2018

Article to be submitted to a special issue of the Journal of Peer Production on the topic of  
“The Institutionalization of Shared Machine Shops: New Spaces, Networks and Practices.”

## **Abstract**

In this paper we present results from an 18 month-long online-based ethnography of Project Ara, in which Google managed to enroll thousands of voluntary contributors into the development of a modular smartphone. Our argument is that, within this tension-laden firm-community entanglement, the figure of the “independent developer” emerged as the central mode of organizing development work. In order to demonstrate this point, we make use of the double notion of ‘figure’ and ‘configuration’ which we borrow from Actor-Network Theory and Feminist Science and Technology Studies respectively. We present three sets of practices that were central in configuring the independent developer: first, the techniques used by the company to interest and enroll external developers, second, the design and redesign of development tools that both enable and control their participation, and third, the creative strategies with which these externals inhabit the company-led project. We end by comparing the figure of the independent developer to other modes of organizing work in digital fabrication and suggesting some ways in which it might be re-configured beyond scenarios of pervasive corporate control.

## **Introduction: Digital Fabrication, Firms and Collectives**

In this paper we discuss the “independent developer” as a mode of organizing work that emerged in the complex entanglements of top-down and bottom-up approaches to digital fabrication. We build our account on the case of Project Ara, in which Google managed to enroll a large number of non-company members to voluntarily and creatively contribute to developing a modular smartphone. Our main argument is that, within this project, the independent developer played a central role in ordering the ambivalent relationships between companies, digital design tools, and a large number of unpaid developers. The figure helped to temporarily fold together practices of grassroots development and the organizational control of work and thus obscured the tensions between them. By analyzing how the independent developer was constructed, we aim to recover three of these ambivalences which characterized work in Project Ara and are arguably typical of current entanglements of large firms and grassroots production communities in digital fabrication.

We situate our account within the larger debate surrounding the emancipatory potential of

digital fabrication. In both social sciences and popular press, the rise of digital technologies such as computer-aided-design software or 3D printing is often connected to hopes of more democratic modes of production (Anderson 2012; Ferdinand, Petschow & Dickel 2016; Raymond 1999; von Hippel 2005). Hackerspaces, FabLabs, and other community-based design and manufacturing projects, are seen as offering a revolutionary chance to alter power and labor relations (Benkler 2006). However, the increasing engagement of large firms with these spaces and communities casts doubt on their transformatory potential. While bottom-up movements remain important, large companies try to shape and exploit their voluntary contributions in various ways (Söderberg & Maxigas 2014). This might take the form of introducing different kinds of organizational openness to innovation processes (Chesbrough 2003), establishing relationships with communities (Dahlander & Magnusson 2005), or fostering innovation platforms or ecosystems (Gawer & Cusumano 2014; Ferdinand & Meyer 2017). These entanglements of bottom-up and top-down approaches to digital fabrication give rise to new modes of organizing work. However, when judged against the high hopes of democratization originally linked to digital fabrication, these often appear deeply contradictory. On the one hand, a large number of people are enabled to creatively participate in the development and production of various goods. On the other hand, corporations find new ways of controlling their labor and appropriating its results.

Against this backdrop, Google's attempt to enroll thousands of voluntary contributors into the creation of a modular smartphone is a prime example of the entanglement of large firms and community practices in digital fabrication. More specifically, Project Ara represents an attempt to transfer the platform approach, well known from software marketplaces like the Android or Apple App Stores, to the domain of hardware development. While a small team within Google partnered with a number of other companies to develop the phone's basic unit, developing functional hardware modules (Fig. 1) was left to external developers [1]. Google's hope was that they would create a large number of unique functional modules like gamepads, night-vision cameras, or medical devices in order to make the company's platform more attractive to a wide range of users. In this way, the figure of the independent developer was critical to Project Ara's staggering goal of creating an aesthetically and functionally customizable smartphone that was, as Google put it on their website, "designed exclusively for 6 billion people." [2]

As detailed in the next section, we conceptualize our analysis of the independent developer by drawing on the double notion of figure and configuration as developed in Actor-Network Theory (Akrich 1992; Latour 1992; Woolgar 1991) and feminist Science Studies (Castañeda 2002; Haraway 1997; Suchman 2007, 2012). We believe it is useful to combine these approaches, because they help us to write from the different perspectives of the many actors involved in organizing work in digital fabrication without resorting to a simple top-down/bottom-up dualism. Viewing the independent developer as a figure, we refer to it as at once the *effect* of distributed practices in the socio-material network of Project Ara and a *mode of ordering* the elements of that network in relation to one another. This allows us to observe that, even though Project Ara is initiated by Google, different human and non-human actors contribute to the emergence of what comes to be seen as the naturalized figure of the independent developer. By unpacking the figures constitutive elements and the powerful practices through which they are “figured together—or *configured*” (Suchman 2012, p. 49), we can recover the ways in which it orders the relationship between practices of grassroots development and organizational control of work.

In order to unfold the way in which the independent developer emerged as the dominant mode of organizing work in Project Ara, our paper is structured as follows. After briefly elaborating on the main tenets of our theoretical conception and methodology, we will focus on three ways in which the figure of the independent developer folds together practices of grassroots development and the organizational control of work. We discuss how Google, a large, profit-oriented company, picked up on ideas of grassroots development and indeed started a major hardware project relying on the voluntary contributions of non-company members. We further show how the digital fabrication tools supplied by Google enabled the creative participation of thousands of developers, while simultaneously working as a way of controlling their activities. Lastly, we draw attention to the way in which the often invisible work of voluntary contributors sustained and shaped the company-led Project Ara, and how their exit from the project was connected to its eventual failure. We close the paper with a discussion of the independent developer’s main characteristics as a mode of organizing work in digital fabrication. We will show how it compares to other modes (e.g. the employee, the crowdworker, the user innovator) and point to some ways work in Project Ara could have

been organized differently.



Figure 1: A prototype of the Ara phone with customized modules (author: Maurizio Pesce)

## **Users and Cyborgs: ANT and Feminist Approaches to Configuration**

The notions of figure and configuration, as we intend to use them originate in Actor-Network Theory (ANT) and Feminist Science Studies during the 1980s and 90s. In contrast to how these two streams of work have been commonly read, we argue that for analyzing the complex entanglements of large corporations and communities of external developers, they are best taken together. On a fundamental level, both approaches are useful for analyzing such new constellations, because they allow us to think about all actors and power as effects of networks of material-semiotic relations. This enables us to avoid simple dualisms between top-down and bottom-up modes of organizing as well as human actors and non-human means. On the level of conceptual repertoires, we find it useful to combine ANT and feminist approaches because they allow us to bring out different aspects of the relationships between corporation and external developers in Project Ara. Early studies in ANT offer us the conceptual language to describe the construction, stabilization, and orchestration of such relational networks (Callon 1986; Latour 1988; Law 2012[1987]). Feminist scholars criticized these studies for overemphasizing the position of powerful actors like scientists and technology designers (Star 1991). They suggested a different conceptual repertoire that

enables us to engage with the agency of marginal actors and their potential to resist, subvert and hybridize. To make this point clearer, we will briefly sketch out how the notions of figure and configuration are positioned within these larger frameworks before showing how we combine them in our analysis of the independent developer in Project Ara.

Work in early Actor-Network Theory was concerned with the question of how non-human actors, specifically technological objects, orchestrate the socio-technical networks into which they are inserted. One important way to answer this question was the turn to a relational semiotic vocabulary that allowed it to talk symmetrically about human and non-human actors (Akrich & Latour 1992). It was in this vein that Steve Woolgar suggested the idea that developing a new technology included what he called “configuring the user” (Woolgar 1991). Working against the metaphor of “machine as text,” he proposed that designers, or writers, always oriented their development activities towards anticipated users, or readers, of the technology. Configuring the user included “defining the identity of putative users, and setting constraints upon their likely future actions” (ibid., p. 59). What was materialized in both the physical shape of the machine and its accompanying contracts and instructions was then not the user as a concrete individual, but the user in a semiotic sense, constructed through the designers’ activities. In a similar turn, Madeleine Akrich and Bruno Latour noted that a big part of any innovator’s work was that of defining and inscribing into the artefact a certain vision about the world in which it was to be inserted (Akrich 1992). Of course, such scripts or programs of action were never truthful representations of “the user in-the-flesh” (Latour 1992), especially when they were put into new contexts. What Akrich (1992) captured in the notion of de-description was that actual users might ignore the script, enact it in unanticipated ways, or even change the artefact itself. While both approaches have been widely influential within Science and Technology Studies, work on user configuration and scripts has been picked up most notably in what came to be called User Studies (Oudshoorn & Pinch 2003, 2008). These studies focused, among other things, on the different techniques used by organizations to construct an idea of who the user might be (Akrich 1995; Oudshoorn, Rommes & Stienstra 2004). More recently, this discussion on user representation was opened up to include the cultural work that goes into the very production of people as users as well as the productive activities of these users within the processes of design and production (Hyysalo, Jensen & Oudshoorn 2016; Oudshoorn 2003).

Feminist Science Studies criticized early ANT for focusing too much on the work of powerful actors like scientists, conquerors, and designers and described those studies as “centered, managerialist, and even military in character” (Law 2009, p. 150; compare Star 1991). These scholars preferred writing from the standpoint of the subjugated, because it was more likely to maintain the contestability and non-innocence of all knowledge without buying into claims of radical relativism and infinite interchangeability. In the spirit of generating “situated knowledges” (Haraway 1988) these scholars developed a conceptual language that helps us to engage with always only partially connected communities of the marginal, the rebellious, and the monstrous. In our concrete case they help us to think from the standpoint of the precarious communities of developers outside of powerful companies.

Of specific relevance here is Donna Haraway’s discussion of the terms figure and figuration (Haraway 1997, p. 11; see Haraway 1991 for her preceding work on material-semiotic actors). She develops a sense of figures as recurring rhetoric or visual tropes that condense and order whole “universes of knowledge, practice and power” (Haraway 1997, p. 11) in necessarily specific and therefore contestable ways. Much like Woolgar’s user, these are not literal representations of any one concrete entity in the world, but rather “performative images that can be inhabited” by such entities. Figures, thus, are always the product of specific worlds and have world-making effects. It is in this sense that Haraway stresses the “contaminated practice” of figuration (ibid, p. 8) as a political tool. Reading the world through different figures, or maps of practice, is what she proposes to do with both her cyborg and modest witnesses (Haraway 1991, pp. 149–182; Haraway 1997).

Haraway’s work was picked up and extended as a critical tool for both the de-construction of dominant figures and their re-figuration (Braidotti 1994; Hayles 1999; Kember 2003). We want to draw particular attention to the contributions of Claudia Castañeda and Lucy Suchman in systematizing Haraway’s writings in a methodological sense (Castañeda 2002; Castañeda & Suchman 2014; Suchman 2007, 2012). They usefully define figuration as “the simultaneously semiotic and material practices [...] by which a concept or entity is given a particular form” (Castañeda 2002, p. 3). A figure, then, is the material-semiotic effect of these practices. It is an entity which, embodied in technologies, texts, visual representations, or

bodies holds together materiality and meaning. However, figures are neither stable nor identical with any of their material instantiations. As they circulate through social worlds, as they become differently embodied, they remain mutable and generative of new effects and entities. “Figuration is thus understood here to incorporate a double force: constitutive effect and generative circulation” (ibid.). Hence, by following the process of figuration in concrete, situated practices, researchers can recover a figure’s constituent elements as well as their relations and eventual transformation (Suchman 2012). This is precisely what we intend to do in the case of the independent developer of Project Ara.

### **Mapping out the Independent Developer**

The independent developer is the figure we chose to follow through Project Ara. We suggest that the independent developer, as a figure, was shaped through and remained generative of novel forms of organizing work in the entanglement of large corporations and communities of external developers in-the-flesh. By attending to the diverse practices that went into its production and continuous transformation, we are able to recover these relations and map out the material-semiotic network of work in the major digital fabrication effort that is Project Ara. In referring to these practices, we use the term configuration, rather than Haraway’s figuration to emphasize two important aspects of our perspective. On the one hand, our use of the term underscores the active contributions of a large number of actors, not only the designers of a company. The goal is to go beyond the simple binary of top-down and bottom-up approaches by presenting a multi-perspectival account without claiming the purity of either perspective. On the other hand, configuration retains the idea of “double force” in that the process of constructing a figure is consequential for all entities implicated in the process and does not only affect the structurally less powerful.

We will analyze three sets of differently situated practices that contributed to configuring the independent developer in Project Ara. Each of the three sets exhibits a particularly important aspect of the configuration, without claiming to present a complete picture of the events in the project. First, we will discuss the material-semiotic practices used by the company to configure the independent developer. We will draw particular attention to the techniques used to construct an idea of who the independent developer might be and how s/he should relate to other elements of the project. We will also discuss how the company attempted to interest



actual people in their vision of the independent developer as the central element of a democratized mobile hardware ecosystem, thereby enrolling them into a company-led development project. Second, we will focus on the ways in which a range of hardware development tools, provided by Google and its partnering companies, contributed to the configuration of the independent developer. We will show how a particular version of the independent developer was inscribed into the material shape of these artefacts. As an effect, the tools enabled externals to creatively participate in the project while at the same time functioning as tools of controlling their actions. The third part explores how external developers themselves contributed to the emergence of the figure by enacting it in unforeseen ways. The central aspect here is not that actual people inevitably differ from the vision of corporate actors. Rather, we want to emphasize that their active engagement in trying to act as independent developers elaborated and transformed the figure constructed by the company.

### **How to Follow the Independent Developer through Project Ara**

Our account builds on a variety of empirical materials, reflecting the diverse practices and actors involved in configuring the independent developer. Central to following the practices of the developers community was an 18-month-long online-based participatory observation of a group of external developers who took part in Project Ara from its official launch to shortly before its termination. The group, consisting of about ten people, granted us full access to their meetings, documents, and internal communication channels. In concrete terms this meant that in addition to conducting several interviews we were a visible part of the group, affectionately referred to as “the study guys.” We joined a series of video conferences, read the posts and comments in their private online forum, studied the documentation they produced, and took part in communications within the group and between the group and other Ara developers. Partly due to the online-based character of the group, all of these activities were readily recorded, which allowed us to access them later and structure them for our analysis. In the course of our engagement, our role gradually changed from silent observers to active participants. Especially in the later phase, we used this position to share our perspectives and learnings with the group.

To substantiate our account, we draw on the broad range of other data that the particularly open character of Project Ara made available. This included public statements by Google and

other actors (such as media and external developers), the phone's technical documentation, recordings of talks and conferences as well as public discussions that were posted on the official website, open forums, and social media platforms. Additionally, we conducted several interviews with the official coordinating team at Google as well as other involved companies and participated in one of the official developer conferences organized by Google. This broad and rich collection of data made it possible to trace the course of the project from its very early stages until its end. It allowed us to enrich our account of Project Ara by studying it from multiple perspectives.

For the analysis, interviews and large parts of the video conferences were transcribed, coded, and together with various field notes ordered along a timeline. By doing this, we were able to trace the re-configurations within Project Ara, how the project evolved over time, which ideas were dropped and which could stabilize. We paid attention to the activities of various actors, including individual external developers, emerging developers groups and communities as well as Google, partnering firms, and the technical and organizational system they developed. This procedure helped us to avoid overemphasizing Google as the central actor in the project. As we will show in the following sections, the path taken was not a linear one, but rather one characterized by the contingent and often conflicting perspectives and activities of various actors.

Our analysis focuses on some of the most important socio-material relations folded together in the figure of the independent developer and leaves out others. It is important to stress that the figure *as an object of empirical research* is delineated, by us as researchers, from the larger (though finite) universe of practice and significance of which it is a part. Thus, even as we speak of the independent developer as a mode of organizing work in Project Ara, we realize that the figure is not first conceived of within this context. In fact, it stems from a much larger domain of practice which is already patterned by asymmetric distributions of meanings, knowledges, and resources. These are not themselves explained within the main part of our analysis, but are simply treated as context.

### **Invoking the Figure**

In this section we examine the central role of the company in configuring the independent

developer. We argue that Google's development team uses a number of techniques to construct an idea of who the independent developer might be and what role s/he should occupy in Project Ara (Akrich 1995; Oudshoorn, Rommes & Stienstra 2004). Some of these practices involve testing the idea with actual developers. Determining to what extent actual developers will meet the company's idea of the independent developer works as a way to prove the viability of the project and helps to guide its further development. Following these internal activities as well as unforeseen developments outside of the company, Google presents its vision of the independent developer to a larger audience of people, trying to enroll them into the company's development project.

The inception of the independent developer is closely tied to Google's effort of constructing Project Ara as a hardware analogue to software development. Even though Google's team claims that Ara is a highly innovative, first-of-its-kind moonshot project, it builds on the well-established cultural repertoire of modern technoscience. This becomes very clear in the many ways in which Ara draws on engrained ways of organizing and innovating and is in fact framed as a typical Google project. The most consequential of these references to already established practices is Google's explicit strategy to model Project Ara in analogy to its highly successful software platform Android. Working by analogy allows the company to formulate expectations about otherwise uncertain elements of the project, specifically the external developers. The following quote by project lead Paul Eremenko is especially enlightening in this regard:

“Project Ara is about opening the mobile hardware ecosystem. It's about making the creation of mobile hardware more like the creation of mobile apps. By lowering the barrier to entry. By increasing the number of participants in the ecosystem. By enabling developers to sell directly to the consumer rather than having to go through an OEM [Original Equipment Manufacturer] and by giving developers new hardware design tools that are free and make hardware design more like software development.” (Paul Eremenko, 1st Developers Conference, 09/15/2014)

The analogy drawn by Eremenko does a lot of work in determining who the independent developer might be and how s/he might be positioned in relation to the company and its design tools. First, much like the development of software apps, the development of Ara modules should be accessible to a large number of people, requiring of them little prior experience or resources. Second, externals are positioned as independent of large companies, not only in the development of modules, but also in terms of their production and sale. Third, this independence is crucially enabled by making use of free hardware design tools which, to

be sure, are provided by Google.

However, even as the translation from software to hardware design offers a first way of envisioning the position of the independent developer within Project Ara, uncertainty remains high. Precisely because such a large-scale attempt at digital fabrication has no direct precedent in the realm of mobile phones, it is unclear whether all the actors will come forth to play their roles as anticipated. This is especially true of the external volunteers who are supposed to embody the independent developer, since they are by definition beyond the company's direct control. Hence, Google sets out to test the viability of the independent developer with actual people. This takes the shape of a series of hackathons which are held over a period of six months across the US. In these events, a total of 212 participants, often engineering students, are invited to develop hardware appliances for an altered version of an existing smartphone, containing additional hard- and software interfaces (see Figures 2 and 3).

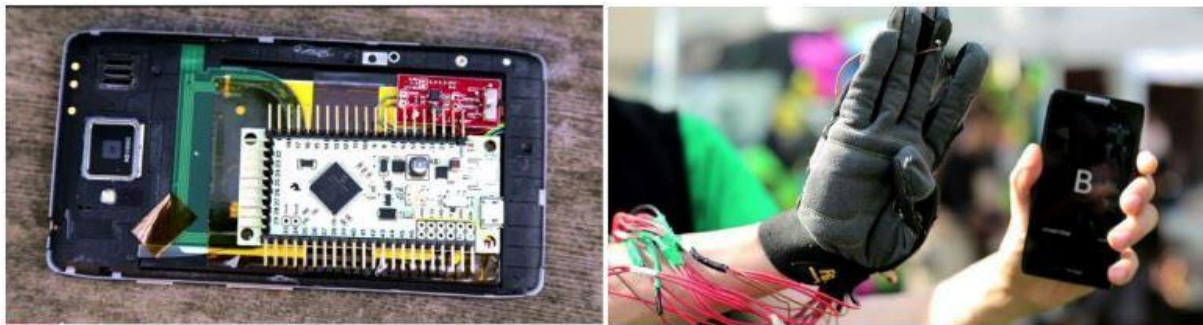


Figure 2: Prototype of a hackable phone [3]



Figure 3: One device developed during the hackathons [4]

Eremenko explained this tool and the rationale behind it as follows:

“It was [...] a hackable version of an existing phone that we loaded up on a truck full with state of the art 3D printing and rapid prototyping equipment, traveled around the country [...] and held makathons in a 48 hour format. And we wanted to see: What would the ecosystem produce around an open hardware platform? [O]ur purpose, wasn't to productize. Our purpose wasn't necessarily to make modules. Our purpose was simply to explore what kinds of things people would create. It was an existence proof if you will.” (Paul Eremenko, 1st Developers Conference, 04/15/2014)

According to Eremenko, the makathons function as a way of testing out the kind of network that would emerge around a “hackable” phone, without incurring the cost of having to build the actual Ara phone. Specifically, it was a way of seeing whether someone would come forth to inhabit the figure of the independent developer and if so, who these people would be and what they would create with the tools provided by the company. The existence proof invoked

by Eremenko refers then as much to the figure of the independent developer as it does to its ordering effect on Project Ara at large. Thus, the makathons can only prove the viability of “the ecosystem,” if there are people to stand as instances of the independent developer, if these people can produce hardware applications with the provided tools and in the allotted time frame, and if their products are judged desirable by the company.

It should be clear, then, that the figure of the independent developer is brought into existence through specific practices of ordering the network elements, like company, tools, and volunteers, in relation to one another. However, as the figure holds these practices together, it simultaneously becomes generative of a new mode of organizing work. Hence, while the independent developer is configured through the efforts of the company, including the provision of certain design tools, the way in which it is taken up by external developers also affects the company’s further work. The following quote by the Google employee in charge of organizing the aforementioned hackathons nicely illustrates this point.

“In many ways, going across the country to the nation’s top universities and just ordinary kind of makers and just ordinary people gave us an early glimpse of the kind of things that you guys would care about, like how do we need to create this modular architecture, what kind of interfaces do we need and all of that.” (Dan Makoski, 1st Developers Conference, 04/16/2014)

Beyond ordering the work of the actors already involved in Project Ara, the figure of the independent developer also generates new relations that would not have existed in a conventional corporate development project. One of the most consequential of these is formed in the company’s engagement with a community-based mobile hardware project called Phonebloks. Phonebloks was a grassroots design project that, like Project Ara, aimed at the creation of a modular smartphone. Started in parallel to Google’s then still not-publicized project, Phonebloks gained enormous popularity in different social-media, generating a large community of enthusiasts and supporters. The great success of the project caught the attention of Google’s internal team and led them to adjust their plans for Project Ara in general and their vision of the Independent Developer in particular. Viewing Phonebloks as an opportunity to jumpstart their own project, Google’s team decided to open up their work in order to enroll the emerging community into the company-led Project Ara. In the first blog post mentioning the project, Google announced the cooperation with Phonebloks, framing it as the necessary complement of their own work:

“We’ve been working on Project Ara for over a year. Recently, we met Dave Hakkens, the creator of Phonebloks. Turns out we share a common vision: to develop a phone platform that is modular, open,

customizable, and made for the entire world. We've done deep technical work. Dave created a community. The power of open requires both. So, we will be working on Project Ara in the open, engaging with the Phonebloks community throughout our development process [...]" (Official blog post, 10/29/2013)



Figure 4: Hakkens' design for a modular smartphone: Phonebloks [5]

Successfully enrolling the Phonebloks community concretizes the company's idea of what the independent developer could look like. At the same time, this new relationship creates a situation in which the company can no longer directly control all activities in Project Ara [6]. The company needs to make sure that the newly found external developers are not only interested in the project, but indeed enrolled into the network (Callon 1986) in a way resonant with the figure of the independent developer. One important way of doing this is inviting the community to a "developers conference" consisting of a series of talks and lectures on all aspects of the project from industrial design to sales. A major part of these presentations lay out Google's idea of the independent developer and its role within Project Ara. By continuously addressing the audience as module developers and by underscoring their central role in creating the revolutionary modular smartphone, the speakers try to enroll the attendees into the position envisioned for them by the company. In fact, Project Ara's innovative potential is framed as resting entirely on the creative work of external developers:

"What new things can I do that I couldn't do before or that I can't do today? Those of you in this room here today and everybody else joining us online you are going to be the ones who are going to answer this question. [...] You! You are gonna do it by the modules that you develop, by the modules you create. You! Now it's not gonna be without frustration, [...] it's not gonna be without a lot of hard work, but it's you that are gonna make it happen and answer that question." (Kaigham Gabriel, 1st Developers Conference, 04/15/2014,)

The conference's big success, reaching a total of 10,000 people on- and offline [7], marks the point in time at which the figure of the independent developer begins to circulate more

widely. While the vision of enrolling voluntary contributors was first discussed within the company's internal team and then tested with a small number of externals, the conference can be viewed as Google's attempt to enroll a large number of new actors into the socio-material network that is Project Ara. Hence, the company occupies a central position in configuring the independent developer and bringing the figure to life. Google's designers use different techniques to construct an idea of the independent developer's identity and position within the project. They then also circulate the figure to externals in hope that it will be picked up in practice.

### **Tools of Control**

The use of the development tools provided by Google is a constitutive part of configuring the independent developer. Without those tools, producing compatible modules is seen as virtually impossible specifically for people that, like many external developers, lack prior experience in hardware design. However, while the tools enable participation, they also regulate it by setting parameters for their user's behavior (Woolgar 1991). As they allow the design of some modules and hinder that of others, the tools can be viewed as an attempt by the company to control the contributions made by external developers. Yet, even though the tools materialize the asymmetric power relation inherent in the figure of the independent developer, such inscriptions (Akrich 1992) can be contested and potentially changed in future iterations.

The tool at the heart of Project Ara's ambitious digital fabrication plans is a software package by the name of Metamorphosis. The freely available toolkit is supposed to free developers from the need to build physical prototypes, thus making the design of Ara modules more like the creation of software apps. In order to do this, Metamorphosis includes tools for everything from designing the circuit board to performance testing, pricing and ordering modules. In the words of Ara's project lead Eremenko, this is supposed to "alleviat[e] multiple design-build-test-redesign iterations," [8] and would guarantee the production of modules that are "in essence correct by design." [9] To allow this, Metamorphosis is programmed with all the design rules and standards Google has developed. Standardizing module development in this way is crucial for a project that integrates the distributed activities of developers who are by definition removed from direct organizational control. Even slight deviations, for instance in

the physical dimensions of a module, can lead to incompatibility with the phone's main platform, rendering the device non-functional. While the use of tools is thus an indispensable part of the figure of the independent developer, the specific way in which they embody the standards and design rules set by Google also inscribes into the figure the asymmetric power relations between the company and external developers.

The software's user interface illustrates how the development tools both enable and control participation. By presenting the developers with a simple drag-and-drop function for optimally positioning and connecting electrical components, the software allows people with little prior experience in hardware design to create functional modules. On the other hand, this limits the number of building blocks and leaves developers little room to manipulate their inner workings. As one of the Metamorphosis employees explained to a crowd of externals during a conference:

“So, instead of working with tiny little components, thousands of connections, millions of lines of code, you work with larger blocks that encode the details such as you have heard on all the presentations throughout the day. All the details, they are very, very necessary for the system to work, but not necessary for you to see at every step you design.” (Theodore Babty, 2nd Developers Conference, 01/14/2015).

In essence, the software embodies Google's view of the position of the independent developer within Project Ara. While the tool makes it relatively easy to create a module, the software sets strict parameters for *how* to do so, thus enforcing the design rules set by Google. Also, by not offering any way to change those rules, the tool reiterates the division of labor on which the project is based. Everything relating to individual modules should be done by the independent developer, whereas everything relating to the overall architecture of the phone is done by the company.

While this may sound like a perfectly feasible strategy to guarantee Google's dominion over Project Ara, in practice things are more complicated. This is because, even though the tools mediate the design rules set by Google in a relatively rigid way, the rules themselves do not remain static. In fact, the tools need to be constantly updated to reflect both the creative module ideas created by actual external developers and the architectural improvements made by the company. These changes can be traced nicely in the different versions of Google's official guidebook for module development, the Module Developer's Kit (MDK). This



document is freely available on the Internet and contains specifications regarding module size, material and layout, as well as 3D models of reference modules, including their software code. As Eremenko proudly remarks:

“[I]t [...] happens to be the first open reference design for a smartphone that’s completely freely available on the Internet. So, all of these schematics, all of the drawings and all of the code that we have to date is linked from that URL.” (Paul Eremenko, Tech Conference, 09/15/2014)

However, as actual developers try to work on their modules, they often find that the current version of the rules does inhibit some of their more creative ideas. In such a case, developers often try to contest the rules, asking the company to change them in the next release of the MDK. This tendency is illustrated nicely in the case of “Flippypad,” a concept for a game-controller-module that garnered considerable attention in the early stages of Project Ara (Figure 5). Despite its enormous popularity among developers and press, Flippypad did not adhere to the rules set in the MDK and could therefore not be realized within the larger framework of Project Ara. Yet, when requests from external developers piled up, Google signaled its willingness to *change* the rules, to permit designs judged desirable. The official Q&A page of the Project Ara website stated:

“Q: Are modules that join two endos or attach a flip screen to an endo supported by the MDK?  
A: This is not currently allowed by the industrial design language in the MDK. However, the whole purpose of getting a very early MDK draft out was to get developer feedback and adapt. We think some of the concepts out there are pretty cool. And to the extent that they don’t compromise other aspects of the design, e.g., structural integrity, we will try and make sure they are supported by the platform.” (Project Ara website, accessed 04/24/2014)

This episode illustrates the point that, while the power relation inscribed into the tools is asymmetric, it is neither static nor one-sided. The company might use tools to shape the actions of external developers, but developers can also contest those inscriptions, at times pressuring the company into changing the rules in future iterations. In this sense, even though the tools are a constitutive part of the figure of the independent developer, enabling externals’ participation and setting parameters for their activities, the specific way in which they do so is constantly changing as the project progresses.



Figure 5: Flippypad, a creative module concept that could not be realized within the tool's parameters (author: Samuel Herb)

### Inhabiting Project Ara

In this section we examine the way in which external developers themselves contribute to the emergence of the figure of the independent developer. We want to emphasize three aspects of this process. First, the independent developer becomes an attractive way for actual developers to think about their own work. By embodying the figure in their development activities, they help to put Project Ara into practice. Second, developers “in-the-flesh” never fit the image in all regards. This creates problems for their participation in the project, but it also elicits creative responses on their part, which further elaborate what it means to be an independent developer (Star 1991). Third, while the figure of the independent developer is imbued with asymmetric power relations between Google and the externals, there is no guarantee that people will continue to inhabit the figure. When organizational changes and technical difficulties arise, external developers cease their voluntary contributions and the project as a whole begins to crumble (Callon 1986).

The independent developer, initially envisioned by Google, is put into action when it is picked up by actual developers [10]. By enacting the figure, they decisively contribute to the initial momentum of Project Ara. This is illustrated by the success of the first Developers Conference hosted by Google. The company later reported 6.800 attendants, 10.000 downloads of the Module Developers Kit and 2660 applications for development hardware [11]. In fact, we find that the way Google rhetorically constructs the independent developer is very appealing to a broad range of people. Externals start identifying with the figure for

different reasons ranging from the chance of a monetary profit, to fun, to altruistic motives. As the figure begins to circulate through different media, it gains a performative quality (Akrich 1992). It does a lot of work in enrolling thousands of very differently situated enthusiasts into Google's project. At this early stage, forums are filled with posts like the one below that link being a voluntary contributor in Project Ara to various personal hopes and desires:

"I'd really love to get involved in the development of Ara modules as I believe it is an incredible engineering feat with a great cause behind it. Empowering the next 5 billion is a staggering goal, but I believe it can be done. The majority of people in my country do not use smartphones and I'd like to assist in developing modules for their needs." (External developer, private forum, 04/20/2014)

However, it soon becomes apparent that for many people there is a mismatch between their ideas and wishes and the independent developer constructed by Google. While people "love to get involved" for one reason or another, most of them lack the knowledge and resources to do so. Remarkably though, this does not result in people questioning the vision of the independent developer or leaving the project altogether. Instead, they engage in the often invisible work of finding ways to inhabit the figure nonetheless (Suchman 2016). On the one hand, this further obscures the asymmetric power relations implied by the mode of organizing work that is the independent developer. On the other hand, these efforts make it possible for voluntary contributors to pursue their own agenda within a company-initiated project.

The first of these elaborations is the departure from Google's original image of the independent developer as someone working individually. Realizing that they cannot pursue the development of Ara modules alone, people identifying as independent developers try to overcome this problem by organizing groups to pool resources and share knowledge. One of the members of the group we followed summarizes this process in a forum post:

"During the Ara developers conference, it became clear that there were a number of people that would like to either learn more about developing a module or collaborate on the development of one. However, due to a number of varying restrictions, knowledge or access to resources for example, they were unable to do this. This group, now known as [Alphamod], was started during those discussions and here we all are. " (External developer, private forum, 04/18/2014)

The fact that they can only *become* independent developers *as a group* has consequences for how they organize their work. Responsibilities need to be distributed, goals negotiated, video conferences attended. In effect, this means that a lot of the voluntary contributions of external developers to Project Ara does not take the form of developing modules for the smartphone, but of sustaining and coordinating groups of enthusiasts scattered all around the globe. Much

of the work they do is that of *becoming* independent developers by reducing the misfit between themselves and the figure. It is this invisible work (Suchman 2016) that allows the company's project to move forward, while simultaneously allowing external developers to pursue their own interests.

Interestingly, it is possible to follow how the mutual shaping between the figure and the people that stand as its instantiations ripples through the material-semiotic network in which they are embedded. One striking example is its effect on the actual modules that the group develops. While the Ara smartphone was pitched as a revolutionary device that could incorporate all sorts of innovative functionality, our group decided to develop a module that would be *as simple as possible*. Hence, instead of developing the sophisticated hard- and software necessary for something like a night-vision camera, glucose meter or game controller, our group decides to develop a simple button module (literally a module with a physical button on it) without any particular functionality at all. One central reason given for this by the developers is that starting from something so easy would both consolidate the group and facilitate learning for interested individuals.

“If we started off with something very, very basic [...] that would give us that chance to create a working relationship, a way of working together, a structure and pass on that basic knowledge to as many people as possible.” (External developer, videoconference, 04/26/2014)

While creating a basic module is still very much an attempt at a meaningful contribution to Project Ara, it is clearly at odds with the innovative work that Google had intended external developers to do. We can observe here that, even though Google clearly occupies a powerful position in configuring the independent developer, the company's control over actual developers always remains limited. As a figure like the independent developer will always be in need of elaboration in the practices of the people inhabiting it, behavior cannot be inscribed into it “in anything like a complete or coherent form” (Suchman 2012, p. 56).

In fact, there is no way for the company to guarantee that people will continue to embody the independent developer. When the gap between the figure as envisioned by Google, its material instantiations (e.g. in the form of development tools) and the abilities or interests of developers “in-the-flesh” becomes too great, the latter can simply stop their contributions and abandon the project. In the course of Project Ara, several events make developers doubt

whether building modules will be technically feasible and whether Google remains committed to building the project in a way that facilitates the meaningful participation of external developers. For instance, even though the independent developer is constructed as someone who uses the tools provided by Google, there are continuous delays in making available both the development hardware and software. Under these conditions, a member of our group concluded in an interview, “some of the more technically minded guys had nothing to focus on, and perhaps as a result they started losing a little interest.” Perhaps even worse than the lack of appropriate tools is the fact that Google does not continue to demonstrate and prove their commitment to the external developers as they did in the earlier stages of the project. Following a leadership change within Google’s internal team, public statements become rare, one of the developers conferences is canceled and questions by externals remain unanswered. The following forum post by an enraged enthusiast epitomizes the growing doubt of many developers:

“What is the story...ProjectAra Insiders [sic]...Are you going to read the forum? Are you going to respond? Are you going to answer questions? Are you going to acknowledge contributors? Are you going to facilitate small independent developers? Are you going to be truth tellers? Is Google going to do the "right thing". "Do no evil?" I hope so...” (External developer, open forum, 08/20/2014)

Eventually, most development activity ceases. As the figure of the independent developer becomes ever more difficult to inhabit, most people leave Project Ara long before it is officially discontinued by Google. Commenting on the end of the modular smartphone, the final post in our group’s internal forum reads “Awww, man. And since all the good-will and enthusiasm has gone after they "went private" I doubt anybody else could pick up where they left off” (external developer, private forum, 09/02/2016).

### **The Independent Developer as a Mode of Organizing Work**

We have shown how the figure of the independent developer is configured within the material-semiotic network that encompasses the company, the tools, and external developers “in-the-flesh.” We have also tried to illustrate how the emerging figure orders the activities of the various actors involved in Project Ara. We will now widen the scope of our analysis beyond this particular case to link the independent developer back to the universe of material-semiotic practices to which it belongs. These are the ambivalent entanglements of large firms and grassroots movements in the realm of digital fabrication and the new working relations they create. In order to do this, we will compare the independent developer to some other modes of work which we typically encounter in the space of digital fabrication. This should

serve as a way of contextualizing our findings and summarizing the independent developer's main features.

Let us start from the perhaps obvious but important difference between the independent developer and the employee. It should be clear from the above discussion that these two map out sharply distinguished ways of organizing work. In contrast to the employee, the independent developer implies a non-contractual relationship between the company and developers. The latter do not become members of the organization and are not paid by it. This position of externality extends not only over the development of a product, but also includes production and sale. At the same time, externality in terms of membership and payment does not mean independence in all respects. On the contrary, as opposed to earlier instantiations of the figure, for instance in the gaming industry, the independent developer of Project Ara is only conceivable with reference to a company's development project. It is at least partly configured by the company, which both identifies and enrolls actual developers, and attempts to specify parameters for their action within the project.

The mode of work implied by the independent developer crucially depends on the development tools provided by the company. These tools at once enable the participation of externals with little prior experience or resources and set parameters for their action. However, the relation between external developers and tools is constructed differently than in the case of crowdworkers (Kleemann, Voß & Rieder 2008) or prosumers using parametric mass customization tools (von Hippel & Katz 2002). Instead of carrying out rather atomized micro-tasks or choosing from a range of predefined parameters, external developers are expected to use the tools in creative ways. While the tools are designed and re-designed to enable such creativity, changes to the tools are ultimately made by the company to guarantee the compatibility of modules and platform. Thus, unlike the (digital) artisan (coons 2016), the open source developer (Gläser 2006, p. 264), or the maker (Toombs, Bardzell & Bardzell 2014), externals can neither change nor create derivatives of their tools.

Finally, with regards to the relationship it configures between developers and their products, the independent developer is not any version of the user (Woolgar 1991), the lead user (von Hippel 1986) or user innovator (von Hippel 2005). In the logic of the figure we describe,

people develop their products not for their own use, but for the use of others. In fact, modules are supposed to be produced for sale on a platform market regulated by the company and similar in style to the Android App store. In this sense, the independent developer bears some resemblance to the classical figure of the entrepreneur (Schumpeter 1947). The main difference is that this is not a case of creative destruction, but of unpaid, creative labor that benefits the established firm by elaborating its platform. A smartphone “exclusively designed for 6 billion people” [12] ultimately benefits the company that controls the project.

Although, in the end of our story, it does not. Project Ara was officially discontinued in September 2016. We do not intend to explain why the project failed. We do, however, believe that there is something to be learned from Project Ara when analyzing other digital fabrication projects. As the entanglement of large companies, voluntary contributors, and developer communities become increasingly common in such projects, it is worth considering how work in these new constellations is organized and how it could be organized differently.

### **Re-Configuring the Independent Developer?**

When it becomes clear that its promise of empowerment and openness cannot be so easily converted into a practical reality, external developers stop contributing to Google’s project. But is this all that can be done in the end? To end on such a note would leave us with a rather bleak outlook on Project Ara and similar projects like it. Instead of ending there, we would like to stay true to the more visionary dimension inherited in our notion of configuration: The possibility to imagine how things could be otherwise. Engaging in this practice of re-configuration or what Braidotti calls the “practice of ‘as if’” (Braidotti 1994, p. 6), we want to close our account of the independent developer by suggesting a way in which work on a modular smartphone could have been, and could still be, organized differently. The point is not to offer a definitive solution to the tensions between practices of grassroots development and corporate control. The point is to remind us that there is nothing inevitable about the way Project Ara turned out and work in digital fabrication could be organized differently.

Indeed, we find moments of potential re-configuration throughout the project and originating both inside and outside of it. Here we will simply point to two such moments to illustrate how work in Project Ara could have been organized differently beyond pervasive corporate control on the one hand and voluntary developers ceasing their contributions on the other. One of

these potential re-configurations lies in the developers' departure from Google's vision of a highly innovative smartphone in the very act of developing modules. We mentioned earlier that, instead of building a highly sophisticated module that would fall in line with the company's expectations, "our" group of developers decided to build a module that would be as simple as possible. While this was done partly because the further development of Project Ara was not predictable, and its technical specifications were still provisional, the approach to do something simple points to a different mode of organizing work as well. Importantly, the simplicity of the module was viewed as an opportunity to learn how to work together *as a group*. The goal was to use Project Ara to build a community, to connect to other people sharing the same interests, and to spread the basic knowledge and skills of soft- and hardware development beyond the project itself. In short, this episode can be read as a vision of appropriating a company-initiated project to build a way of working and learning together beyond the goals of the company itself.

A second challenge to the dominant mode of organizing work in Project Ara originated from outside of the project. Dave Hakkens, the founder of Phonebloks, envisioned a very different relationship between companies, design tools and voluntary contributors and therefore a very different mode of organizing development work. In Google's vision of Project Ara both the creation of ideas and the actual development of modules would be done by unpaid externals using the company's design tools to ensure compatibility. In stark contrast, in Hakkens' vision of Phonebloks, only the ideas would be created by the community while the technically challenging work of building the actual modules would be done by companies according to newly established industry standards. In essence, whereas Project Ara was a company trying to enroll external developers, Hakkens envisioned a community of enthusiasts, trying to enroll companies. His idea, then, was not so much that of individual independent developers producing for a company's platform market, but that of a powerful community of users that could actively influence the industry's R&D efforts to realize their creative ideas. Phonebloks reminds us that meanings and materialities can be figured together in vastly different ways. There is nothing that inevitably binds modular phones or other high-tech products to a platform logic or to the control of a single company.

## **Conclusion**



We presented Project Ara as an example of organizing work in digital fabrication and the complex entanglements of large companies and developer communities that are common in such contexts. In order to go beyond the simple dualism between bottom-up and top-down perspectives, we used the concepts of figure and configuration. We showed how different actors contributed to the emergence and temporary stabilization of the ambivalent figure of the independent developer and how that figure in turn became the dominant mode of organizing work in the project. The independent developer was crucial for building the material-semiotic network of Project Ara because it allowed Google to interest external, unpaid developers into Project Ara and to enroll them into a very specific position by providing tools that enabled and controlled their contributions. At the same time, the independent developer was brought into existence in the contingent practices of external developers that tried and sometimes succeeded in following their own agenda. The independent developer cannot, therefore, be described as the strategic outcome of the activities of Google or any other actor in the network. Even though the figure was crucial for holding the precarious relations between these different actors together and made it possible to continue the project besides its many ambivalences. Finally, pointing out the precarious state of such networks, the notion of (re-)configuration reminds us that work in digital fabrication could always be organized differently.

## **Acknowledgements**

We would like to thank Tobias Rixen and Uli Meyer who made our study of Project Ara possible and contributed to an earlier version of this paper. We also thank Sabine Maasen, the editors and two anonymous reviewers for their insightful comments that helped us to improve the final product.

## **Notes**

[1] We distinguish the term “external developers” from our notion of the “independent developer” and both from Woolgar’s notion of “the user”. The term “independent developer” was prominently used by various actors in the field to denote individuals who voluntarily contributed to the company’s project by developing, but not necessarily by using products. We use the term in an analytical way to refer to a performative image of organizing work, or

what we call a figure. We use the term “external developers” to speak of the developers “in-the-flesh”, the concrete people who contributed to module development in Project Ara.

[2] <http://www.projectara.com/>, accessed 09/12/2015

[3] <https://www.youtube.com/watch?v=aSSuIceLP2g&t=0m39s>, accessed 07/03/2017

[4] <https://www.youtube.com/watch?v=aSSuIceLP2g&t=0m54s> ; accessed 07/03/2017

[5] [https://commons.wikimedia.org/wiki/File:Phonebloks\\_open.jpg](https://commons.wikimedia.org/wiki/File:Phonebloks_open.jpg) , accessed 07/31/2017

[6] Indeed, we argue that Phonebloks configured an entirely different mode of organizing work, which was written out of corporate accounts as the project progressed. We will return to this point below when addressing possible re-configurations of the Independent Developer.

[7] <https://www.youtube.com/watch?v=IQhbM55F23U&t=0h43m00s>, accessed 07/31/017

[8] <https://www.youtube.com/watch?v=uj9AcXJ54QQ>, accessed 02/19/2018

[9] [https://www.youtube.com/watch?v=v2OEKL1w\\_\\_4](https://www.youtube.com/watch?v=v2OEKL1w__4), accessed 02/19/2018

[10] Like the members of the developers group that we joined during our research, a typical developer in Project Ara can be described as an amateur in regards to the development of mobile phones. Typically, s/he (though mostly he) had some background and expertise in software and sometimes hardware design and was interested in the Project because of Google’s initial promise of democratizing hardware design by providing free tools and lowering the barriers to entering the project.

[11] Paul Eremenko, statement at tech conference,  
<https://www.youtube.com/watch?v=IQhbM55F23U&t>, accessed 2018/02/27

[12] <http://www.projectara.com/>, accessed 09/12/2015

## References

- Akrich, M 1992, 'The De-Description of Technical Objects' in *Shaping technology/building society. Studies in sociotechnical change*, eds WE Bijker & J Law, MIT Press, Cambridge, Mass., pp. 205–224.
- Akrich, M 1995, 'User Representations: Practices, Methods and Sociology' in *Managing Technology in Society. The approach of Constructive Technology Assessment*, eds A Rip, TJ Misa & J Schot, Pinter, London, New York, pp. 167–184.
- Akrich, M & Latour, B 1992, 'A Summary of a Convenient Vocabulary for the Semiotics of Human and Nonhuman Assemblies' in *Shaping technology/building society. Studies in sociotechnical change*, eds WE Bijker & J Law, MIT Press, Cambridge, Mass., pp. 259–264.
- Anderson, C 2012, *Makers. The New Industrial Revolution*, Random House Business Books, London.
- Benkler, Y 2006, *The Wealth of Networks. How Social Production Transforms Markets and Freedom*, Yale University Press, New Haven, London.
- Braidotti, R 1994, *Nomadic Subjects. Embodiment and Sexual Difference in Contemporary Feminist Theory*, Columbia University Press, New York.
- Callon, M 1986, 'Some elements of a sociology of translation: domestication of the scallops and the fishermen of St Brieuc Bay' in *Power, action and belief: a new sociology of knowledge?*, ed J Law, Routledge, London, pp. 196–223.
- Castañeda, C 2002, *Figurations. Child, Bodies, Worlds*, Duke University Press, Durham, London.
- Castañeda, C & Suchman, L 2014, 'Robot visions', *Social Studies of Science*, vol. 44, no. 3, pp. 315–341.
- Chesbrough, HW 2003, *Open innovation. The New Imperative for Creating and Profiting from Technology*, Harvard Business School Press, Boston.
- coons, g 2016, *Something for everyone. Using digital methods to make physical goods*, Unpublished work.
- Dahlander, L & Magnusson, MG 2005, 'Relationships between open source software companies and communities. Observations from Nordic firms', *Research Policy*, vol. 34, no. 4, pp. 481–493.
- Ferdinand, J-P & Meyer, U 2017, 'The social dynamics of heterogeneous innovation ecosystems', *International Journal of Engineering Business Management*, vol. 9, 1–16.
- Ferdinand, J-P, Petschow, U & Dickel, S (eds.) 2016, *The Decentralized and Networked Future of Value Creation. 3D Printing and its Implications for Society, Industry, and Sustainable Development*, Springer International Publishing, Switzerland.
- Gawer, A & Cusumano, MA 2014, 'Industry Platforms and Ecosystem Innovation', *Journal of Product Innovation Management*, vol. 31, no. 3, pp. 417–433.
- Gläser, J 2006, *Wissenschaftliche Produktionsgemeinschaften. Die soziale Ordnung der Forschung*, Campus Verlag, Frankfurt am Main, New York.
- Haraway, D 1988, 'Situated Knowledges. The Science Question in Feminism and the Privilege of Partial Perspective', *Feminist Studies*, vol. 14, no. 3, pp. 575–599.
- Haraway, DJ 1991, *Simians, Cyborgs, and Women. The Reinvention of Nature*, Routledge, New York. Available from: <http://gbv.eblib.com/patron/FullRecord.aspx?p=1195818>.
- Haraway, DJ 1997, *Modest\_WitnessSecond\_Millennium. FemaleMan@\_Meets\_OncoMouse™. Feminism and technoscience*, Routledge, New York, London. Available from: <http://www.loc.gov/catdir/enhancements/fy0647/96012174-d.html>.
- Hayles, K 1999, *How We Became Posthuman. Virtual Bodies in Cybernetics, Literature, and Informatics*, The University of Chicago Press, Chicago, London.
- Hyysalo, S, Jensen, TE & Oudshoorn, N 2016, 'Introduction to the New Production of Users' in *The New Production of Users. Changing Innovation Collectives and Involvement Strategies*, eds S Hyysalo, TE Jensen & N Oudshoorn, Taylor and Francis, New York, London, pp. 1–42.

- Kember, S 2003, *Cyberfeminism and Artificial Life*, Routledge, London, New York.
- Latour, B 1988, *The Pasteurization of France*, Harvard University Press, Cambridge, Mass., London.
- Latour, B 1992, 'Where Are the Missing Masses? The Sociology of a Few Mundane Artifacts' in *Shaping technology/building society. Studies in sociotechnical change*, eds WE Bijker & J Law, MIT Press, Cambridge, Mass., pp. 225–258.
- Law, J 2009, 'Actor Network Theory and Material Semiotics' in *The New Blackwell Companion to Social Theory*, ed BS Turner, Wiley-Blackwell, Chichester, pp. 141–158.
- Law, J 2012 [1987], 'Technology and Heterogeneous Engineering. The Case of Portuguese Expansion' in *The Social Construction of Technological Systems. New Directions in the Sociology and History of Technology. Anniversary Edition*, eds WE Bijker, TP Hughes, T Pinch & DG Douglas, The MIT Press, Cambridge, Mass., London, pp. 105–128.
- Nachtwey, O & Staab, P 2015, 'Die Avantgarde des Digitalen Kapitalismus', *Mittelweg 36*, vol. 24, no. 6, pp. 59–84.
- Oudshoorn, N 2003, *The Male Pill. A Biography of a Technology in the Making*, Duke University Press, Durham, London.
- Oudshoorn, N & Pinch, T (eds.) 2003, *How Users Matter. The Co-Construction of Users and Technologies*, The MIT Press, Cambridge, Mass.
- Oudshoorn, N & Pinch, T 2008, 'User-Technology. Some Recent Developments' in *The Handbook of Science and Technology Studies*, eds Hackett, Edward J. Amsterdamska, Olga, M Lynch & J Wajcman, The MIT Press, Cambridge, Mass, London, pp. 541–566.
- Oudshoorn, N, Rommes, E & Stienstra, M 2004, 'Configuring the User as Everybody. Gender and Design Cultures in Information and Communication Technologies', *Science, Technology, & Human Values*, vol. 29, no. 1, pp. 30–63.
- Raymond, E 1999, *The Cathedral & the Bazaar. Musings on Linux and Open Source by an Accidental Revolutionary*, O'Reilly, Sebastopol.
- Schumpeter, JA 1947, 'The Creative Response in Economic History', *The Journal of Economic History*, vol. 7, no. 2, pp. 149–159.
- Söderberg, J & Maxigas 2014, *Book of Peer Production*, NSU, Göteborg.
- Star, SL 1991, 'Power, technology and the phenomenology of conventions: on being allergic to onions' in *A Sociology of Monsters. Essays on Power, Technology, and Domination*, ed J Law, Routledge, London, New York, pp. 26–56.
- Suchman, L 2007, *Human-Machine Reconfigurations. Plans and Situated Action*, Cambridge Univ. Press, Cambridge.
- Suchman, L 2012, 'Configuration' in *Inventive Methods. The happening of the social*, eds C Lury & N Wakeford, Routledge, London, New York, pp. 48–60.
- Suchman, L 2016, 'Making Work Visible' in *The New Production of Users. Changing Innovation Collectives and Involvement Strategies*, eds S Hyysalo, TE Jensen & N Oudshoorn, Taylor and Francis, New York, London, pp. 125–135.
- Toombs, A, Bardzell, S & Bardzell, J 2014, 'Becoming Makers. Hackerspace Member Habits, Values, and Identities', *Journal of Peer Production*, vol. 5.
- von Hippel, E 1986, 'Lead Users. A Source of Novel Product Concepts', *Management Science*, vol. 32, no. 7, pp. 791–805.
- von Hippel, E 2005, *Democratizing Innovation*, MIT Press, Cambridge, Mass.
- von Hippel, E & Katz, R 2002, 'Shifting Innovation to Users via Toolkits', *Management Science*, vol. 48, no. 7, pp. 821–833.
- Woolgar, S 1991, 'Configuring the user: the case of usability trials' in *A Sociology of Monsters. Essays on Power, Technology, and Domination*, ed J Law, Routledge, London, New York, pp. 58–99.