PRESS PLAY: ACTS OF DEFINING (IN) FLUID ASSEMBLAGES

JOHAN REDSTRÖM
UMEÅ INSTITUTE OF DESIGN
UMEÅ UNIVERSITY
JOHAN.REDSTROM@UMU.SE

HEATHER WILTSE
UMEÅ INSTITUTE OF DESIGN
UMEÅ UNIVERSITY
HEATHER.WILTSE@UMU.SE

ABSTRACT

Although design continuously has been expanding its scope of concern and intervention from products to processes, experience, and entire product and service ecologies, 'things' remain central to how we think about design and use. But 'things' have changed. Contemporary materials, technologies and contexts of design and use, we argue, now result in 'things' that need to be understood as *fluid assemblages* rather than traditional objects. These often combine a surface-level simplicity of use with dynamic, sophisticated, and hidden back-end complexity.

In order to investigate these issues we consider a simple design case and how it has evolved over time and through technological developments: that of pressing *play* to listen to music. Noting the tendencies in the ongoing evolution, with focus on the simple design element of the 'play' button, we suggest that traditional distinctions between design and use are breaking down. Coming to grips with the materials and ecologies of contemporary design practice thus requires the development of design theory and methodologies that allow us to articulate and bring into focus these significant new dynamics.

INTRODUCTION

For the past couple of decades, we have learned that design has moved from product to process, from object to experience. We have also learned that products can not be designed as if they existed in isolation, but that we need to understand them as part of product and service ecologies (Forlizzi 2008; Stolterman et al 2013), not to mention the intricate social and material fabric of people's lives. Indeed, there are now many bids for what it is that design designs, ranging from physical forms to social innovations. And yet, 'things' remain central to how we think about both design and use. But we need to ask the question of what has become of things. More specifically, we need to look at what defines 'thingness' to us, and how design goes about to design it.

One of the more intriguing paradoxes of contemporary design in general, and perhaps of design in the digital domain in particular, stems from our rather complicated relations to complexity and simplicity. To understand the background of these relations, we need to turn at least back to the days of HfG Ulm in the early 1950's, and the search for clear and functional design of technical objects on one hand, and a growing interest in increasingly complex products and systems on the other. At Hfg Ulm, we see early examples of both the kind of industrial design still today highly influential in the technical domain (think Dieter Rams, one of its students) and the first steps towards making design an interdisciplinary project set up to deal with 'wicked problems' (think Horst Rittel, professor in design methodology). It seems we carry two important ideas from our past: a striving for simplicity and a concern for complexity.

While the striving for simplicity certainly is a concern for usefulness and utility, it is also an aesthetic orientation. Indeed, if there is one well-known phrase that captures the Modernist aesthetic, it would probably be 'form follows function'. To say that it is an aesthetic orientation is not to criticize it, but to suggest that it is a part of a worldview inherently tied to a certain way

of thinking and doing design. Of course, there could be other aesthetic orientations, but at least for industrial design as practiced in the North of Europe, this worldview is something much deeper than one out of many approaches one may choose between when doing a project.

The concern for complexity also has its roots in early Modernism, and the idea that there are important relations between design and society, between individual objects and industrial systems. Even in early examples of industrial design, we see an explicit interest in how objects are related to each other in systematic ways, and how design can help address and make sense of the resulting complexity. Today, a corresponding interest in the complex interactions between people and systems of objects can be seen in areas such as 'the internet of things' and in the development of design approaches for circular economies, networks and social innovation, etc. In other words, a concern for complexity and how to evolve design to address it has been a learning process unfolding since the discipline first came about.

THINGS AS FLUID ASSEMBLAGES

As these two strong trajectories are combined (as they often are), we end up in the paradox that is the nexus of this paper: on one hand, we aim to provide as simple, clear and useful interactions as we possibly can; on the other, we aim to address complexity that is at the verge of what we can grasp. To add to the burden, design also in many cases still aims to build on our common sense understanding of what a 'thing' is. In many cases, the paradox is seemingly resolved: we can appreciate the iPhone as an elegant thing held in our hands and at the same time marvel at the richness of apps and services we can make use of through its interface. In some cases, however, it is also clear that we have not resolved the paradox at all, as what seemed to be a simple set of transparent interactions turned out to also be part of the most advanced of surveillance technologies.

A crucial implication of how the intended user experience is now in many cases being dynamically assembled in runtime is that each thing –as experienced– becomes unique in comparison with others of its type, and in ways that are substantively different from the ways in which any mass-produced thing will become in some ways unique due to the ways and contexts in which it is used. This new type of uniqueness stems from the ways in which networked things can be (re) assembled dynamically according to a practically infinite array of parameters. So a thing is unique not only to a specific person, but also to a specific person at different points in time and space. This is why we suggest that such things are to be considered *fluid assemblages*: assemblages because they are made out of a diverse range of material and immaterial resources both contained within the object as it appears in front of us as well as located elsewhere in the network; fluid because their precise forms are assembled in runtime and thus change continuously.

Not only does the thing present itself slightly differently each time due to the dynamic and contextual dependance on various parameters, it might be somewhat unpredictable from a user's point of view also in other ways. If personalities can be attributed to things (Reeves, & Nass 1996; Giaccardi et al 2014), it might be said that some things have a severe case of split personality disorder. Or, to put it in Goffman's (Goffman 1959) terms, the persona that they perform on the front of the stage is quite different from what is going on backstage. So an application may seem like the most attentive assistant while actually monitoring a user's activities in order to package her data and sell her attention to advertisers.

This appears to be a design paradox worth unpacking. To start doing so, we have selected a very simple act of use: that of pressing play to listen to music. Tracing this simple act through a series of examples, we aim to analyze how this striving for simplicity in combination with a concern for complexity risks us causing a rift in how we think and do design. Putting more in motion than just usability, we argue that such rifts are problematic also in the sense that they might undermine the basic social contract between design and use, and thus the basic trust we need to have in the things we use.

PRESSING PLAY

While techniques for music playback have quite a long history, there has been in just the past decade or so an explosion in the development of technologies and services for listening to music. Yet, even with such a diverse array of music playing technologies historically and in terms of currently available options, some things remain constant. We approach a music-playing thing because we want to hear music; and whether we find it on the plastic button of a tape or CD player, the click wheel of a classic iPod, or in the interface of a digital app, we know to look for the familiar right-facing triangle icon. To hear music, we press *play*.

This simple act of pressing play has remained quite consistent, even as the complexity of the underlying systems that make the playing possible has increased tremendously. Indeed, many digital music players are now only one component of vast ecosystems including digital service providers, musicians, record labels, advertisers, and other digital platforms. Moreover, they participate in these ecosystems in much more complex and dynamic ways than their simpler historical

Record player, iPod and CD player. Images by author.

predecessors. This development of contemporary music players thus embodies precisely the tension between simplicity and complexity that we wish to explore.

Specifically, we will here consider simply the assembly or activation of a music-playing thing such that its functionality of pressing play is made available to us. We will trace the nature of this assembly through a variety of cases, beginning with older analogue technologies and watching for significant changes as we move on to address more contemporary technologies and (eco)systems.

EXAMPLES

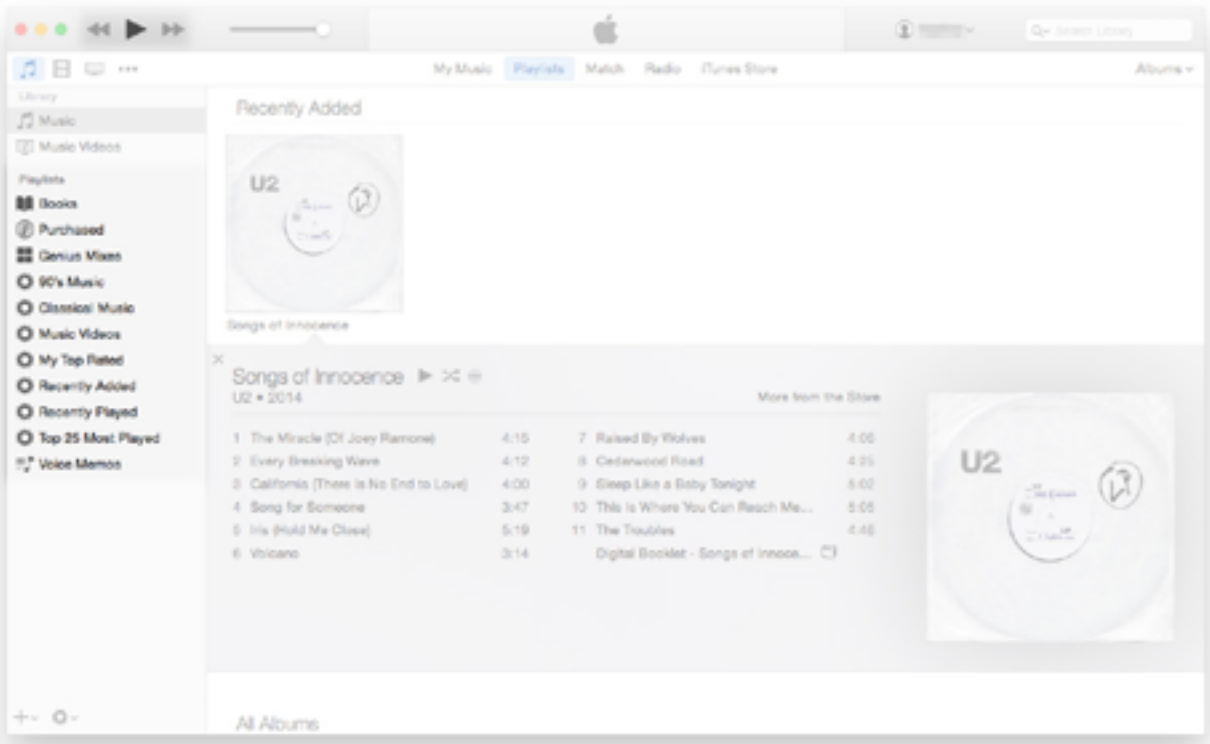
Beginning with pre-digital examples of music playing technology, we can think of classic record and tape players. These devices are quite respectable design objects in the classic sense: things that are mass produced in factories and then purchased by users who then own and can do whatever they like with them. Playing music on them requires plugging in the power cord and perhaps pushing a power button, loading a record or tape, and then pressing play. Especially in the case of a record player, it is typically possible to

Winamp. The familiar music player control panel is now combined with a variety of content sources.

manually intervene at more or less all stages of the mechanical process as well, such as manually lifting and positioning the arm with the needle onto the record, stopping or slowing down the rotation with the hand, etc. All of the assembling of these things occurs on the manufacturer's side before they reach the end users, and unless he or she decides to physically modify the device it will remain the same. Although the mechanics involved are somewhat harder to inspect, this logic also applies in the case of stand-alone CD players.

However, when we move to considering a CD played in a computer rather than a stand-alone CD player we notice some different dynamics emerging. First, and most obviously, computers do much more than play CDs; this is only one of many functions they have which are managed by the underlying operating system and installed software. And software is in fact needed to play the CD. The physical CD drive may not seem much different than that of a classic CD player, but the fact that it is now operated by software marks a key shift. There are now multiple software options that can be used in conjunction with the same CD drive—ones which can be updated and configured independently of the underlying hardware, and thereby change overall functionality. Also, and in contrast to the elegant simplicity and functional transparency of a record player arm lifting up and over a record, loading a CD in a computer launches countless computational processes that are not generally visible (although it might be possible to use system monitoring tools in order to see some of what goes on). This would often include a query to a music database service like GraceNote in order to retrieve the track names for the CD, revealing another (networked) component of the assemblage.

CDs can also be ripped and stored in a computer hard drive, marking another key development when music can be stored in digital formats and played without the need to load an external storage device. And of course they can also just start in digital format and be distributed without the need to ever involve physical storage media other than computer hard drives. As music-playing things, digital music player applications were a rather new kind of animal. They are 'assembled' from a variety of components when the application is launched. These include the code for the application itself and the underlying operating system that manages its processes, including sound output (which might be internal or peripheral speakers). The computer itself

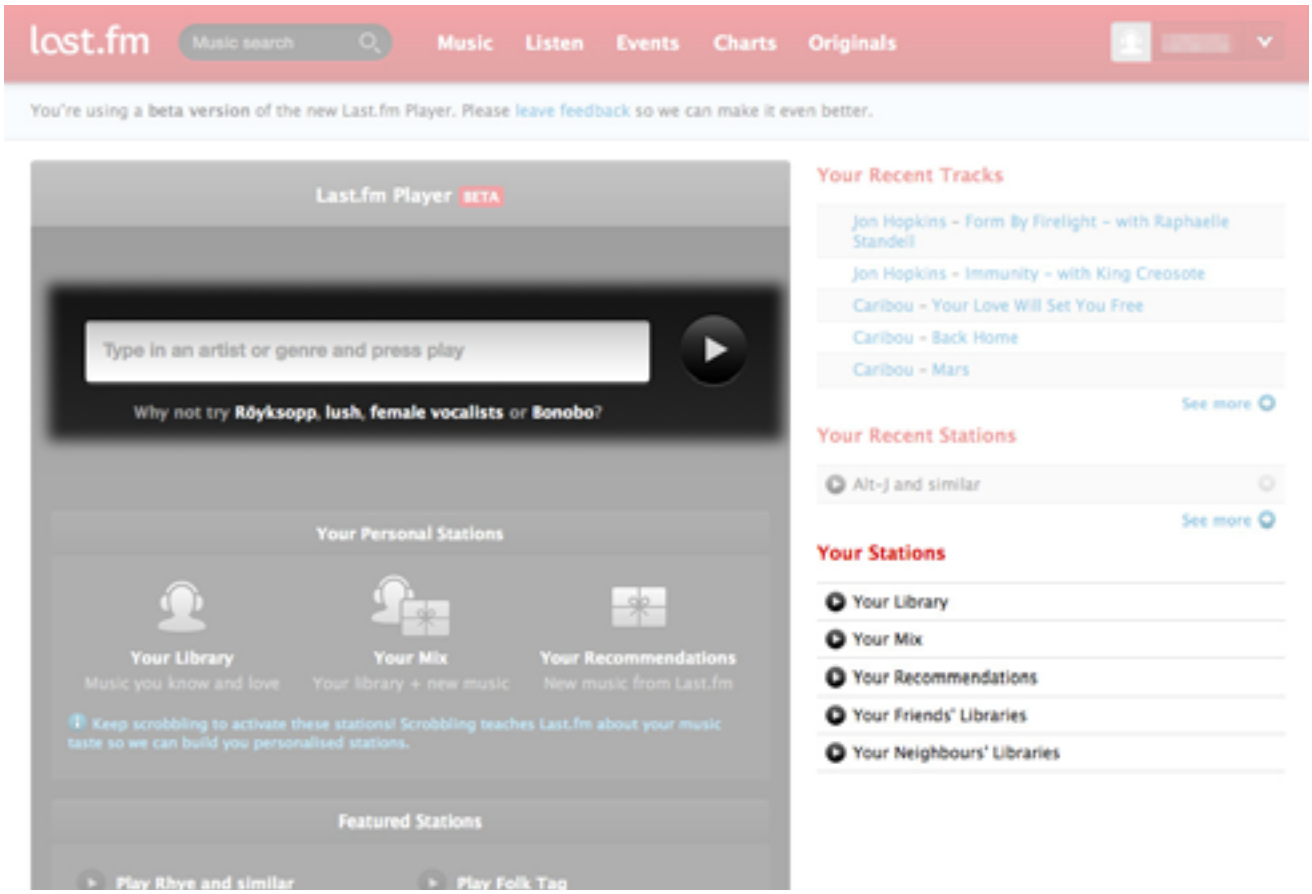


iTunes 12. 'Smart' playlists based on track metadata and usage have become standard.

is the component that seems most object-like in a traditional sense; yet even with this simple example we can see that only a small part of its functionality is determined through its assembly in a factory. Much of what a computer does is rather determined by its operating system and applications, which can be updated and configured in ways that can greatly change functionality without changing the underlying hardware.

In addition to the assembling that happens through basic software, it is also possible to further modify an application's 'assembling' as a thing available for use through configuring its settings. These can change how it behaves and how it looks. One such example are various forms of automatically generated play lists, ranging from 'shuffle' functions first known as 'random' playback order in CD players that mix up the predefined playback order, to more elaborate algorithms based on categorizations, tags and other kinds of metadata attached to the song. Another example might be how the classic Winamp player allowed for customisation through 'skins'. Moreover, anyone could develop these skins and make them available for others; and branding them with one's logo could become a point of pride for their creators, serving as a visual reminder that it was another person (and not Winamp) who created this particular component of the user's personal Winamp assemblage.

At the same time that the MP3 file format and players like Winamp gave people much more freedom in terms of how they could play and distribute music, other trajectories sought to restrict the ways in which people could acquire, listen to, and distribute music, even as they also capitalized on the possibilities of the digital. The most significant player in this regard is arguably iTunes, with its 'walled garden' approach to providing a coherent and seamless user experience while also ensuring that only certain kinds of 'acceptable' use are possible. It is well known that the possibility to make infinite duplicate copies of music files without loss of quality and to easily and widely distribute them via the internet posed a significant challenge to existing structures in the music industry, and led to the emergence of new sociotechnical configurations that is still ongoing. However, even as these dynamics have driven the development of many of the more contemporary music playing systems we discuss, our concern here is with the ways in which these are assembled and appear as things available for use.

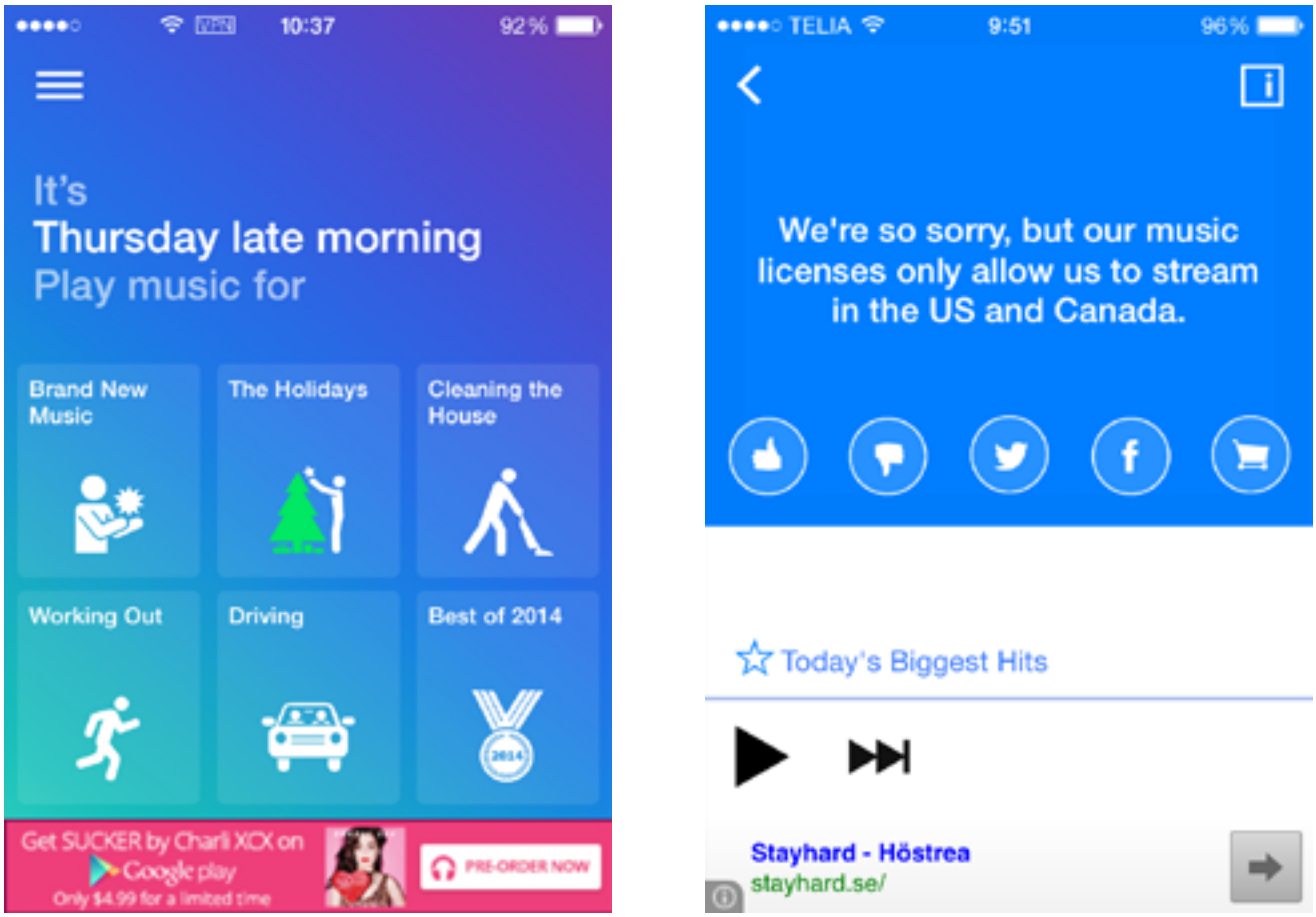


Last.fm. The 'why not try' suggestions below the search bar and the 'featured stations' are customized by user account or, when not logged in, change every time the page is loaded. http://www.last.fm

Another key development associated with iTunes and the iPod music player was in structured metadata associated with media files. This was clearly visible in the iPod in particular, where music could be accessed in multiple ways, through artist, album, genre, playlist, etc. Significantly, iTunes also included metadata reflecting usage, such as play count, skip count, and date last played, as well as data about when the file was added to the iTunes library and last modified. This arguably marks the beginning of the evolution of music players in which usage affects the future constitution and behavior of the system. This can be seen in a single track itself that has updated metadata, and in the resulting ways in which tracks are displayed when sorting according to these variables. However, it also works in a more subtle way by affecting the frequency with which tracks are played on 'shuffle' mode in both iTunes and synchronized devices (such as the various iPod models and now the iPhone).

Both this personalisation of the iTunes data and experience, and the enforcement of certain usage restrictions, are accomplished through accounts. Accounts have now become quite common and effectively extend the relationship between producer and consumer for as long as use of the product continues. Even web-based music players that do not require accounts track users and customize the offerings in fairly sophisticated ways. For example, the Last.fm music player web page (http://www.last.fm/listen) loads a variety of trackers, beacons, and analytics that, as of this writing, include ones for Audience Science, BlueKai, ClickTale, DoubleClick, Google Analytics, Omniture (Adobe Analytics), Qualtrics, Spotify Embed, and Yahoo Analytics. Refreshing the page or connecting from different locations also updates the musical suggestions provided. However, the extensive and fluid assemblage of Last.fm is rather disguised by an interface that invites the user to simply 'type in an artist or genre and press play'.

A significant aspect of this continuing relationship between providers and users is that 'use' can be precisely scripted and either enabled or limited in dynamic ways. For example, use can be customised or restricted based on location. On a basic level, detecting the country from which a person is connecting to a web-based service allows for language customisation and for presenting what is most popular in that country. But it can also be used to restrict access to certain content



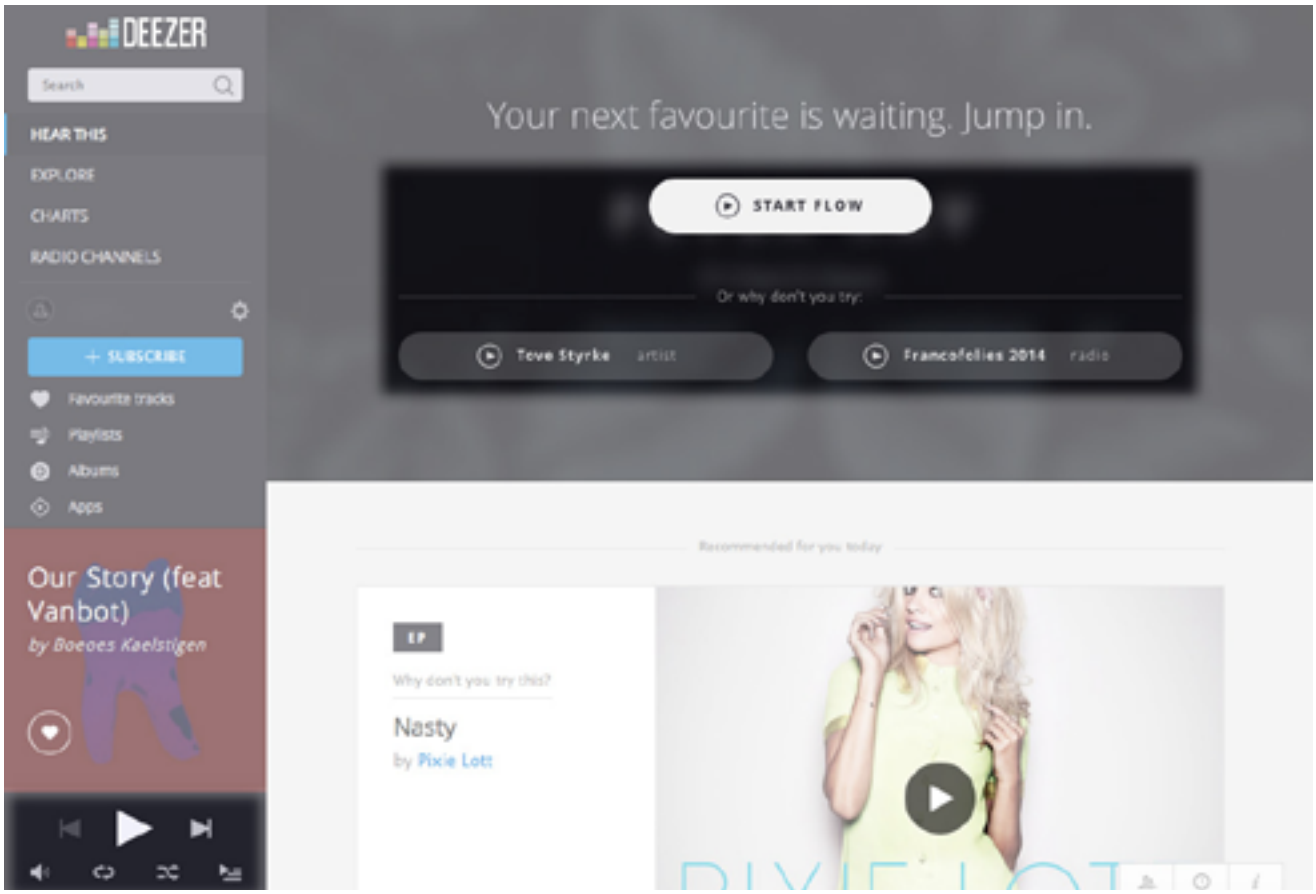
Songza. Ads are loaded with the music player—even when it is not possible to stream content due to geographic location of the IP address in use. http://songza.com

or prevent access entirely, as in the case of Songza that cannot be accessed through internet connections coming from outside the US or Canada. These restrictions can, however, be bypassed by connecting through a VPN service—another component that can be brought into the assemblage on the side of 'use'.

It is interesting to note that when we reach this situation of dynamic customisation there is no longer any single, stable 'object', that can be viewed 'objectively'. Instead, what is stable across users is the set of rules and processes governing the ways in which the product is constituted at runtime for specific accounts connecting from certain locations at certain times—although even these rules themselves change over time. Indeed, one of the most prominent aspects of modern web-based music players, such as Deezer, Slacker, Pandora, etc., is how they adapt their music recommendations over time based on what individuals listen to and indicate that they like.

Importantly, these 'things' can also be continuously disassembled. For instance, streaming content providers may stop making certain content available. Starting to use Spotify on one device will stop playback on another device. Content may also stop being available because of changes in the governing legal contracts, as when the music of an artist from one day to another is no longer available as a new commercial agreement could not be reached. One can also experience the geographical specificity of such legal agreements when traveling, as some content is available in some countries but not in others.

A major dynamic in this runtime production and customisation of music-playing things is that not only are they assembled dynamically, but the components assembled come from a variety of sources. One way this can be seen is in the many examples of services that load ads in conjunction with the application. These ads themselves represent the complex and extensive assemblages of advertising services, such as Google ads or Apple's iAd program. From a slightly different angle, many services now allow for authentication through social media accounts (like those of Facebook or Twitter) and also connect to the functionality of these accounts in other ways (e.g., loading Facebook friends into a 'friends' list, or enabling the sharing of one's



Deezer. Five different play buttons initiating different kinds of content. http://www.deezer.com

activity). The assembling of the music-playing things and their functionality in these instances is enabled and constituted partially through these other services. Indeed, quite a few reasons behind the particular design of some of these assemblages are related to the shift from a focus on consumer purchases to selling user data in many business models: since what is 'sold' is not a 'thing', but data about the user that can be used to for instance customize advertisement and direct users to certain other services, gathering as much such data as possible becomes a key driver. This is a major reason for the increasing importance of accounts to access music, but it can also be seen in the extensive user profiling and tracking in services not necessarily requiring a login. For example, SoundCloud's cookie policy (https://soundcloud.com/pages/cookies) describes how, in addition to their own cookies used for managing sessions, they use a number of third party services (from Google, Quantcast, ATInternet, Scorecard Research) that provide analytic and advertising functionality. They also use the "similar technologies" of Clear GIFs, Flash cookies, HTML5 local storage, activity tracking ("Localytics" service provided by Char Software, Inc), app performance tracking ("adjust.io" service provided by Adjust GmbH), and bug reporting ("Crashlytics" service provided by Crashlytics, Inc and "HockeyApp" service provided by BitStadium GmbH).

Another general source of input for runtime customisation is users themselves. One way this works is through application settings, but there are also a number of other means by which use of a thing can later feed back into how it is assembled. As previously noted, simply recording which music tracks are listened to can affect how music can be sorted and displayed. This allows for features that display the artists, tracks, etc. that a person has listened to the most. Recent listening is placed front and centre in the Rdio online application, turning activity into the main content of the site in the form of a collection of album artwork representing a timeline of recent listens. And of course listening activity also feeds back into the recommendations provided later on.

Finally, it is interesting to note the extent of the shift from buying something and then really and truly owning it, to using things to which one has access only provisionally. Systems can be upgraded or, more neutrally, modified without users' consent. Content and features can be added and removed. Use is regulated

through a mutually reinforcing combination of system architecture and law (Lessig 2006) such that, for example, customization and restriction of a web-based service based on location is reinforced by stipulations that users must not try to circumvent them. Instead of operating manuals, users are now faced with sometimes staggeringly extensive terms of service (which are perhaps even less likely to be read); and they must accept these before gaining access to the system, thereby entering into standing legal agreements of which most typical users have only the faintest understanding. Yet these terms of service sometimes contain dire warnings and regulations regarding use, such as the Google Play terms of service that states in part (https://play.google.com/intl/en/about/play-terms.html):

"NONE OF THE PRODUCTS ARE INTENDED FOR USE IN THE OPERATION OF NUCLEAR FACILITIES, LIFE SUPPORT SYSTEMS, EMERGENCY COMMUNICATIONS, AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL SYSTEMS, OR ANY OTHER SUCH ACTIVITIES IN WHICH CASE THE FAILURE OF THE PRODUCTS COULD LEAD TO DEATH, PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE."

Pressing play has become serious business indeed.

DISCUSSION

In earlier work, acts of defining what a given thing is was discussed based on a distinction between acts of design and acts of use (Redström 2008). Consider a glass bottle as a typical example. Acts of designing – of making as craft – a glass bottle would be acts such as preparing the material, heating the glass, blowing and shaping it, cooling it, etc. Acts of designing a bottle for industrial production would instead entail acts of producing a prototype that can be mass-produced, through sketches, models, etc. While the process of making the bottle as such can differ, there is still a clear distinction between such acts of defining what the 'bottle' is, and what then happens as we use it. It will still be a matter of defining what the bottle 'is' (to us), but these acts will be based on the fact that the bottle is there for us in its physical form. And so I may use it to contain fluid that I can drink, thus defining it as a drinking vessel, but I can also use it to express my feelings by throwing it to the wall, thus (re-)defining it as a kind of prop in a performance of sorts. It can be used as a small window in a cottage I'm building, thus defining the bottle as a kind of building material. In fact, we might even use it as material for making a new bottle, thus closing the loop. In general, we might say that there are potentially a range of different acts defining what this thing is, but that they basically fall into two categories: ones of 'design' causing the thing to come into being, and ones of 'use' using the thing for some purpose. This distinction, then, is the basis for a kind of social contract established between design and use that, on one hand, allows 'designers' to create objects for intended forms of use and intended users, and then 'users' to acquire, interpret and make use of these objects for their own purposes based on the typically predictable and stable properties of the objects as present physical things in their lives (Hallnäs & Redström 2002).

When it comes to the more 'fluid assemblages' we currently create and use, this basic picture is breaking down. More importantly, the basic social contract between design and use is becoming increasingly problematic as the underlying premisses for that contract are being replaced by new forms of making and using. This causes a complexity, and a rupture, we do not yet know how to address, but that we, as a start, need to try to articulate. The basic cause for this change is that the fluid assemblage is never really made, at least not in the common sense that a bottle is made. The fluid assemblage is continuously in the making, in ways that intertwine acts of design and acts of use over time in ways that traditional mechanical objects certainly can not. Yet, as we tried to show with the examples above, many 'things' do their best in keeping up appearances, maintaining that the basic contract is still valid and that the basic relations between designing and using are still in place.

Looking at contemporary design, there is in many cases no single, uniform, consistent, stable thing when it comes to design objects (Wiltse, Stolterman & Redström, 2015). Rather, as we use computational and other new materials, the composition of things is determined on the fly according to a potentially infinite array of constantly shifting parameters and operations, many of which are hidden. 'Thingness' has so far primarily been defined in either one of two main ways. The first is in accordance with the physical presence, functionality, and qualities of an object itself. The second is, broadly, through the complex and dynamic technical, organizational, and sociocultural networks that bring it in to being, sustain it, and infuse it with meaning. However, as the evolution of devices for listening to music outlined above clearly illustrates, neither the object-centric nor the social constructivist account of what a thing is allow us to describe the more fluid assemblages now being developed, designed and used.

This new kind of complexity we are now facing is something rather different, even as it recasts a classic and related tension between simplicity and complexity in new ways. The complexity that stems from dynamics of use in a social context was in a way external to the

things themselves. The composition of a thing and the composition of the systems in which it was embedded were closely related, but also possible to separate. Now we are in a situation in which the composition of a thing at any given moment is determined in non-trivial ways by variables that are external to the thing itself.

As neither an object-centric definition nor a social constructivist notion of what a thing is seem to capture this new kind of emerging complexity, the next logical step would most likely be to instead ground such a definition in some notion of human experience. Typically when considering user (or, in a slightly more sophisticated take, human) experience, it is this experience that is thought to be dynamic while the object remains stable. Taking such a view it is also, significantly, those aspects of an object that are present to the experiencing human that are thought to be relevant. In the case of computational technologies, the typical goal of interface design is to mediate the complexity of the underlying technology such that the user will experience only those aspects that are necessary for the desired 'experience' or functional goals. This holds true for much interaction design, whether focused on effective usability, rich experience, or something else.

A couple of other traditions have also fed into framing experience as the sine qua non of understanding human-technology relations. One that has gained much traction in recent years is the philosophical orientation of phenomenology, which focuses on the ways in which the world is apprehended by humans. Phenomenology has also become fairly influential as an approach to framing experience as a dimension along which technologies can be considered and analyzed. Another tradition is that of semiotics and, more generally, cultural studies of technologies, in which the significance of a thing is derived from its symbolic resonances and its imbrication in social practices. Taken to an extreme, these traditions suggest that when the ostensive experience of a thing is adequately accounted for, there is not much more to be said about its role in human affairs. However, this is no longer (if it ever was) the case as we attempt to get a grip on contemporary things. On the contrary, relying on user experience as an analytic frame occludes much of what goes on with and through such things, and the significant structural and functional elements that exist beneath the surface of what is perceivable on their user-facing surfaces.

It appears almost as if we are heading towards a blind spot, where certain issues are occluded by our prevalent perspectives, and where we therefore need to develop new accounts of the basic 'what' it is that we design. This leaves us with a problematic gap between existing frameworks and emerging design issues, a gap we believe design theory needs to articulate and new design methodologies need to address. Getting a grasp on these fluid assemblages in order to responsibly design with and within them requires moving beyond the anthropocentric viewpoint of traditional user-centered design—even as it is precisely human experience and integrity that we care about.

We suggest that notions of 'things' as fluid assemblages might be part of the vocabulary needed for such articulations to close this gap between prevalent object-centric, social constructivist and experiential accounts of what a thing is. This is a perspective that resonates with the original Nordic notion of 'things' as political gatherings around shared matters of concern, always in the making (Binder et al 2011; Latour 2004). In fact, it might be said that stable things are giving way to the unfolding of shifting landscapes defined through ongoing processes of 'thinging'.

REFERENCES

Binder, T., De Michelis, G., Ehn, P., Jacucci, G., Linde, P. & Wagner, I., 2011, *Design Things*, The MIT Press, Cambridge, Massachusetts.

Forlizzi, J., 2008, The product ecology: Understanding social product use and supporting design culture, *International Journal of Design*, 2(1), p. 35.

Giaccardi, E., Speed, C., van Allen, P. & Grossen, J.F., 2014, *Companion to DRS 2014: Design's Big Debates*, The things about design: Of ghosts, spirits and material practices. Umeå University, Umeå, p.175.

Goffman, E., 1959, *The Presentation of Self in Everyday Life*, Anchor Books, New York.

Hallnäs, L. & Redström, J., 2002, From use to presence: on the expressions and aesthetics of everyday computational things, *ACM Trans. Comput.-Hum. Interact.*, 9(2), pp. 106-24.

Latour, B., 2004, Why Has Critique Run out of Steam? From Matters of Fact to Matters of Concern, *Critical Inquiry*, 30(2), pp. 225-48.

Lessig, L., 2006, *Code: Version 2.0*, Basic Books,.

Redström, J., 2008, RE:Definitions of use, *Design Studies*, 29(4), pp. 410-23.

Reeves, B. & Nass, C., 1996, *The Media Equation: How People Treat Computers, Televisions, and New Media as Real People and Places*, Cambridge University Press.

Stolterman, E., Jung, H., Ryan, W. & Siegel, M.A., 2013, Device Landscapes–A New Challenge to Interaction Design and HCI Research, *Archives of Design Research*, 26(2), pp. 7-33.

Wiltse, H., Stolterman, E. & Redström, J., 2015, Wicked Interactions: (On the Necessity of) Reframing the 'Computer' in Philosophy and Design, *Techné: Research in Philosophy and Technology*, 19(1).