

Shifting Design Capability to Third-Party Developers: An Affordance Perspective on Platform Boundary Resources

Full papers

Amir Mohagheghzadeh
University of Gothenburg
Amir.mohagheghzadeh@ait.gu.se

Fredrik Svahn
University of Gothenburg
Fredrik.svahn@ait.gu.se

Abstract

Boundary resource theory has emerged as conceptual tool for understanding the complex relationship between platform owners and third-party developers. Drawing on existing theories of boundary objects and boundary spanning competence it suggests that platforms may offer influence over external ecosystems, yet keep them at arm's length. To exercise such governance, however, platform owners have to figure out how to design boundary resource to transfer design capability to third-party developers. Addressing this challenge, we analyze a digital platform initiative in the automotive industry from an affordances perspective. By doing so, we have explored what platform boundary resources allow developers to achieve, rather than what they are. As a main obstacle in the transfer of design capability, we found that platform owners' perceptions of what a specific boundary resource affords often differ from third-party developers understanding of the same resource.

Keywords

Third-Party Developers, Platform, Boundary Resources, Affordance, Design.

Introduction

Digitalization is fundamentally transforming product innovation practices in a wide range of industries. While innovation has been viewed as an internal activity, firms increasingly rely on loosely coupled external actors for continuous supply of novel functions and services (Dougherty and Dunne 2011; El Sawy et al. 2010; Nambisan and Sawhney 2011). In doing so, they draw on digital platforms and platform thinking to empower external ecosystems and, eventually, derive value from them. However, setting up such boundary-spanning innovation environments is a notoriously difficult task as incumbent firms and ecosystems typically are powered by different innovation regimes (Svahn and Henfridsson 2012). That means they form expectations and initiate actions on the basis of different "principles, norms and ideology, rules and decision-making procedures" (Godoe 2000p. 1034). Belonging to "different social worlds" (Star and Griesemer 1989) there is a considerable risk platform owners and platform users form different perceptions of the same platform.

Contemporary platform research offers the concept of platform boundary resources to understand the complex relationship between platform owners and third-party developers. Boundary resource theory (BRT) emerged as a way to deal with the tension between inherent openness and generativity of ecosystems and platform owners' need for control. Drawing on existing theories of boundary objects and boundary spanning competence, BRT suggests that APIs (Application Programming Interface), software tools, regulations, and other platform resources may "serve as the interface for the arm's-length relationship between the platform owner and the application developer" (Ghazawneh and Henfridsson 2010). In other words, platform owner can exercise governance through the platform, without explicit involvement in ecosystem innovation.

BRT offers a powerful model for how to understand platform governance. In particular, it uncovers mechanisms for how platform owners can exercise influence over external ecosystems. However, we find

that the third-party developer perspective is backgrounded – or at least addressed at a meta-level – in extant research. As a consequence, BRT offers little hands-on guidance in how to actually design boundary resources. That is, how to design boundary resources that are “plastic enough to adapt to local needs and constraints of the several parties employing them, yet robust enough to maintain a common identity across sites” (Star and Griesemer 1989, p. 393).

With an ambition to extend BRT we ask how incumbent firms shape platform boundary resources to shift design capability to external actors. While existing research has primarily focused on what such boundary resources are, we have explored what they allow developers to achieve. In doing so, we analyzed a digital platform initiative from an affordances perspective. We were primarily interested in understanding how differences in principles, norms, and ideology play out when platform owners and third-party developers shape their respective perceptions of boundary resources. Therefore, we first followed how a platform owner designed a range of platform resources, on the basis of their own innovation logic and largely anecdotal knowledge on third-party development. We then studied how a group of third-party developers actually used the same resources in the development of new applications.

Platform Boundary Resources

Tiwana et al. (2010) describe a digital platform as “the extensible codebase of a software-based system that provides core functionality shared by the modules that interoperate with it and the interfaces through which they interoperate” (p. 676). While such generic core functions are well defined, a digital platform does not impose specific uses or applications – it is not functionally decomposed (Simon 1996). As a consequence, digital platforms tend to be layered rather than hierarchical (Yoo et al. 2010), which allows for new modules to be recursively added on top of the platform to extend its functionality. In other words, these modules – or “add-on software subsystem” (Tiwana et al. 2010) – can be designed and developed by independent third-party developers in form of software applications. Recent research on platforms has increasingly recognized the significant value of such third-party developers and their contributions to platform innovation (Bergvall-Kåreborn et al. 2010; Bosch 2009; Boudreau 2012; Evans et al. 2008; Messerschmitt and Szyperski 2005; Remneland-Wikhamn et al. 2011).

To capture value from third-party development platform owners often seek to bring numerous developers on board (Boudreau 2012) and to increase variation they strive to mobilize differentiated and uncoordinated audiences (Zittrain 2006, p. 1980). By increasing the platform’s installed base they hope to stimulate a wide range of complementary assets in form of innovative applications and services (Evans et al. 2008; Hanseth and Lyytinen 2010; Messerschmitt and Szyperski 2005), but also to foment network effects among third-party developers and end-users (Gawer 2014). However, platform owners need to figure out how to stimulate and capture value from external developers’ innovation, without constraining creativity. Therefore, a main objective in digital platform design is to facilitate and increase the development process and maintain the integration of complementary assets (Ghazawneh 2012).

Digital platforms make an interface between platform owners and third-party developers. Drawing on existing theories of boundary objects (Carlile 2002; Gal et al. 2008; Star and Griesemer 1989) and boundary spanning competence (Levina and Vaast 2005) recent platform research has suggested that the complex relationship between platform owners and third-party developers may be better understood through the concept of platform boundary resources (Barrett et al. 2012; Eaton et al. 2015; Ghazawneh and Henfridsson 2013b). A key aspects of boundary resource theory (BRT) is its emphasis on the transfer of design capability (Ghazawneh and Henfridsson, 2013, p. 175). Properly designed boundary resources allow platform owners to put specific design capabilities in the hands of third-party developers. At the same time, such boundary resources allow third-party developers to access functionality of the platform, but also to serve end-users through software applications that will be deployed and become part of a platform ecosystem (Baldwin and Woodard 2008; Gawer et al. 2012; Tiwana et al. 2010; Yoo et al. 2010).

According to Ghazawneh and Henfridsson (2013a, p. 174) platform boundary resources are “software tools and regulations that serve as the interface for the arm’s-length relationship between the platform owner and the application developer”. The literature categorizes boundary resources in different ways, including SDKs (Software Development Kit) and APIs, guidelines and documents, agreements and Intellectual Property (Bergman et al. 2007; Yoo et al. 2010). Clearly, designing boundary resources is a difficult task, requiring platform owners to understand developers demand. At the same time, as

emphasized in boundary object theory, platform owners should not strive to close the gap between themselves and third-party developers; such differences empower this model of innovation rather than undermines it. Therefore, boundary resources should be “plastic enough to adapt to local needs and the constraints of the several parties employing them, yet robust enough to maintain a common identity across sites” (Star and Griesemer 1989, p. 393). Thereby, they should be adaptable to different viewpoints, yet robust enough to maintain identity across them (p. 387).

Recognizing this paradox of concurrently bridging and embracing differences between platform owners and third-party developers we emphasize the need to communicate with developers. To successfully introduce new boundary resources, platform owners have to understanding developer demands. Clearly articulated objectives and functionalities of resources allow external developers to contribute better. However, as these resources are designed and develop by the platform owners’ experts, embedded in an existing innovation practice, there is a substantial risk they miss-understand how boundary resources will be used by external developers. We argue that existing BRT has overlooked such differences in perception.

In this research we ask how incumbent firms shape platform boundary resources to shift design capability to external actors. In addressing this research question we put particular attention on the implications of differences in perception. In what follows we present a theoretical framework, based on the concept of affordances. This framework allows us to exploring what platform boundary resources allow developers to achieve, rather than studying what constitutes such resources.

An Affordance Perspective

The notion of affordances was initially introduced by James Gibson (1986) to describe the different ways animals perceive their ecosystem. According to Gibson animals perceive objects, such as a rock, differently depending on its affordances (i.e. the different ways the animal can use the rock). Essentially Gibson suggested that people interact with an object only after they perceive and understand how the object is useful and what it can afford them (Gibson 1986). Based on this logic, Gibson (1986) argued that the properties of an artifact should be defined independently of the artifact per se but rather on how it can be used. In other words, the perceptions of what an artifact can *afford* is equally important as what the artifact is (Treem and Leonardi 2012).

Leonardi (2011) elaborates on Gibson’s concepts of affordances in relation to technology. He uses the affordance concept as a lens to describe the intertwined relationship between material and human agency. According to Leonardi (2011) based on the contexts in which technologies are used, different potentials for action can be seen. Further, as individuals perceive the material properties of an object in a particular (individualistic) way, affordances are unique (Leonardi 2011). So, while people might identify technological properties in a similar way, affordances are unique because they rely on individual perceptions of its possibilities and restrictions. Leonardi (2011, p. 154) elaborates “...as people attempt to reconcile their own goals with the materiality of a technology, they actively construct perceptual affordances and constraints. Depending on whether they perceive that a technology affords or constrains their goals, they make choices about how they will imbricate human and material agencies.”

Platform boundary resources link technology and humans, but they also make an interface between platform owners and third-party developers. As this relationship evolves, new opportunities are created. We seek to capture the dynamics of this interplay by using the term “affordances” in the sense that new combinations of platform features and innovation environments continually create possibilities that affect innovation output. In particular, we apply the affordance lens to explore a platform owners’ perspective in the design phase of boundary resources and compare it with external developers’ perspectives.

Research Design

In our research we have studied Volvo Group, a leading manufacturer of trucks, buses, and construction equipment at the global market. Volvo Group has initiated several research projects over the last couple of years in order to explore the effects of digitalization and the role of digital platforms. As part of one of those projects, initiated early 2013, we set up a qualitative study of Volvo’s attempts to enter the ecosystem around Google’s Android through a new platform initiative. As a critical part of this project, the firm had to figure out how to shift design capabilities to external developers via boundary resources. The

project was guided by the objective “to design, develop, and evaluate a safe open-source connectivity platform concept for infotainment services and applications that satisfy state-of-the-art safety standards.” Over a period of 20 months, starting early 2013, Volvo developed a whole range of different platform boundary resources. For the purpose of this paper we selected 5 of those resources (see table 1) for further analysis; one document (Safe Connectivity Recommendation); one SDK (Occlusion App); and three APIs (Vehicle API, Distraction Level API and Safety API).

In fall 2014, 172 students (30 groups of 5 to 6 students) used Volvo’s new Android-based prototype platform for application development. The assignment lasted for ten weeks and focused on the boundary resources provided by Volvo Group. The students were given the broad assignment to “create innovative and original software applications within the area of safe connectivity”.

We designed our research as a qualitative study to specifically explore external developers’ perspective on designed platform resources and then compare them with initial assumptions of Volvo Group experts. As underlined by Kaplan and Maxwell (2005) qualitative approaches are helpful in exploring this type of new socio-technological phenomena and gaining an understanding from the viewpoints of participants (Kaplan and Maxwell 2005). Given that the first author had immediate access to the Volvo project, with opportunity to closely observe and analyze, our study was implemented as a single case study (Yin 2013).

Data Collection

To understand and describe Volvo’s view on affordances provided by the platform’s boundary resources we relied on 13 interviews with Volvo Group specialists, 5 workshops (focused on preparation and packaging of platform resources), 21 project meetings, and 5 project reports. The data was collected in the period Aug 2013 to Feb 2015. The first author was present in all these interviews, meetings and workshops. All interviews and workshops and 6 project meetings were recorded (totally 26 hours) and transcribed (364 pages). The rest of the project meetings were documented by the first author (47 pages) and the project’s online database including 150 decisions, 55 actions. These decisions and actions, in addition to recorded conversations in the meetings and workshops, helped us understand the process of boundary resource design and the project team’s perception of these emerging boundary resources.

In order to understand and describe how third-party developers perceived the boundary resources offered by Volvo, we implemented an online survey and posed it to students. The survey was designed to measure whether developers acknowledged resources and, if so, how often they had used them. 27 groups (out of 30) eventually developed an application and answered our survey, providing solid data for our study. To render fine-grained narratives on boundary resource usage the first author made participant observations of the groups in action (4 hours). He also made semi-structured interviews with all groups but 14 to specifically ask developers about the affordances they could see in each resource. All interviews were recorded (5 hours) and transcribed (65 pages).

Data Analysis

We used the qualitative data analysis software Atlas.ti to store, display and analyze our data. First, we identified candidates for platform boundary resources and develop a table to display them. For confirmation we then interviewed Volvo Group specialists who were engaged in developing these resources. Interviews, workshops, project meetings, and project reports were then analyzed to understand Volvo’s perception of boundary resources and what they would afford developers. We initiated this action by analyzing a project report, including suggested resources and their expected functionality. Complementary interviews with Global Application SDK coordinator and other senior engineers at Volvo Group reflected the platform owner’s perspectives on expected affordances of the designed resources. In those interviews project team members were asked to elaborate the functionalities and affordances of each boundary resources, with a specific focus on third-party developers. Including the empirical evidence from third-party development, extracted from online survey and semi-structure interviews, we then started to code the data material to foreground affordances of boundary resources. In follow-up interviews developers were asked why and how they used different resources. Following the principles of methodological triangulation (Lee and Liebenau 1997) we then used the online survey to increase the validity of the study by combining quantitative and qualitative data sources.

Results

Developing an experimental digital platform, based on Google's Android, Volvo Group developed a number of platform boundary resources. Those boundary resources were manifested as guideline, documents, SDKs, and APIs to help developers design application. Being an experimental project Volvo did not, at this point, invest in developing legal agreements to regulate business relations. Instead, the firm focused on how to promote safe application development, while concurrently stimulating the generation of innovative functions and services for trucks and busses. To be considered safe in the eyes of Volvo, any application has to be able to handle driver distraction, i.e. being able to guarantee that the driver's attention remains focused on the primary task – driving. In what follows we report on the development of five selected platform boundary resource and reflect Volvo's perceptions and visions behind designing them. We also compare this perspective with developers' perceptions of the same boundary resource and elaborate on recognized affordances.

Safe Connectivity Recommendation

To promote driving safety in the area of in-car infotainment¹, Volvo Group considered it important to provide a guideline for third-party application development. Volvo primarily viewed this as a way to share the automotive industry's long experiences and expertise in reducing driver distraction with third-party developers. This experience was documented in various automotive guidelines², but Volvo understood it had to repackage it to make developers actually using it. Therefore, recognizing the need to be brief and to the point project members developed a 12 page informative document, called Safe Connectivity Recommendation. Basically this document had three main chapters; display design recommendation; recommendations for driver interaction; and finally, application performance test. As an example, the display design recommendation provided suggestions for font size and appropriate space between characters to be displayed on in-car screens. Volvo viewed this recommendation as a way to reduce burden on developers and let them focus attention on creative application design, rather than readability. The second chapter afforded a crash course in driver distraction for external developers without any experience from the automotive industry. It highlighted tasks and functions that Volvo considered inappropriate for use while driving, but also gave examples of functions that had to be restricted or degraded while driving. The third chapter contained information about performance testing and described acceptance criteria. Reflecting back on the process of repackaging existing knowledge in a condensed safe connectivity recommendation a human factor specialist at Volvo Group noted that:

“We [Volvo safety experts] have reviewed all of them [existing design recommendations] and took out the ones [specifications and recommendations] that we thought they were specifically important and of course referred to them and we made our own compilation. [...] we basically transformed it [existing design recommendations] into an easier to understand language and also class them and categorize them [the specifications] differently like providing screen shots, tables and so on.”

The safe connectivity recommendation was introduced to third-party developers (university students) at an introduction lecture. Volvo wrote that the document *“is intended to assist mobile application developers to develop safe and user friendly connectivity”*. Thus, Volvo envisioned that this platform boundary resource would afford developers to design and develop safe in-car applications, with limited focus on driver distraction issues. In addition, it was expected to afford a way to rapidly test applications, as it defined acceptable performance according to automotive safety standards and guidelines.

While Volvo were initially optimistic about the value of this resource, it turned out that 40.7% of the developers had never used this resource after 10 weeks of development. A Student recalled that *“in the first two-three weeks we got some similar information [design recommendation] at introduction lectures, so we backgrounded it”*. Obviously, the introduction lectures afforded similar things in a way that was easier to digest. As a consequence, many developers considered the safe connectivity recommendation of limited value.

¹ Hardware/Software products and systems which provides a combination of information and entertainment.

² Such as NHTSA: Visual Manual NHTSA Driver Distraction Guidelines for in- Vehicle electronic devices

Reflecting on the development process one group of developers argued that *“we looked at them [the safe connectivity recommendations] last week and found out that we almost followed everything without knowing them. It is a good confirmation that we are on the right path.”* Another group said that *“we wanted to see if we hadn’t miss any large recommendation”*. This suggests that they mainly used this resource for self-assessment of applications at the end of the development process. In other words, it afforded output control of applications.

Occlusion App 1.0

Beside the safe interaction recommendation, Volvo decided to develop an SDK which developers could use to assess performance of an application, as outlined in the safe connectivity recommendation. The performance test was defined as; *“85% of individual glance durations [during visual interaction with the application] should not exceed 2 seconds. Mean of individual glance durations should not exceed 2 seconds, task completion should not require more than 12 seconds of total glance time at display and controls (for 85% of sample).”*

As part of the SDK the project team developed an application referred to as the occlusion app 1.0. The application could be run on any android device and blanked the device screen for a certain time, as defined by the acceptance criteria. Thereby, it afforded developers to assess their application’s performance using any Android device. When the screen was ON the user could interact with the screen for 2 seconds, while interaction was blocked when the screen was OFF. If the full task was completed within 12 seconds, the application passed the performance test.

From Volvo’s perspective the occlusion app afforded developers to experiment with different solutions to fulfill performance test criteria. However, the SDK was not delivered to developers from start. As described by one of the groups, this complicated adoption: *“It was hard to plan ahead how to use it when we didn’t get access to the resource [from the beginning].”* As a result, only 56% of developers used this resource, at least once before final launch. Groups who used the occlusion app mainly saw it as a way to self-assess the performance of the developed application. However, according to some teams this SDK also had the potential to afford even more; *“The value of Occlusion App was limited for us. Because we had to go back and change. We did not have enough time to do general changes, but it created value for us as we made some small changes. It allowed us to test our app’s performance.”* Properly exercised, this resource could have afforded them to fine-tune their application before final demo, which is well aligned with Volvo’s original ideas about the occlusion app. Interviewing with groups who did not use this application at all, showed that external developers’ expected this resource to afford them to test the application’s performance based on the acceptance criteria. However, as this resource was not available at the beginning and developers had no clue how complicated would be to use this resource, some teams planned not to use this resource. So they designed their application early and these application often required few visual interactions; *“late notification made us hesitate to use it [Occlusion App 1.0] as we [already] decided what to do to. [...]. So our application needed few visual interactions.”*

Vehicle Data API

In order to stimulate external development of applications for the driving context, Volvo developed a new vehicle API. This API would give developers access to internal sensors and data, thereby affording sophisticated integration with the vehicle. When it comes to assumption of developers’ preferences, Volvo thought working with APIs would be easier and preferable. Our survey, also showed that developers were interested in working with this API. 63% percent used this resource, at least once in their development process. There were three groups decided not to use it, as they were afraid of making development process more complicated. Interviews showed that developers found that this API could notify them if the engine is ON or OFF. This notification could be the basis for defining access to the applications’ different functions while driving. In other words, this API could afford them to define two modes for the vehicle – driving or non-driving – which in turn could enable them to restrict some functions of the application according to Safe Interaction Recommendation document; *“The only thing [Signal] we used in our application was to check that the engine is not on or the person is not driving the car”*.

In addition to this affordance, developers also found that the Vehicle Data API could notify them on the fuel level in vehicle, which they could use to create innovative application. This notification could, for

example, be the basis for suggesting the driver to refuel the tank; *“using Fuel level signal, we try to create value for drivers through suggesting the best place to refuel in the path towards the pre-selected destination.”* In that sense, this API afforded developers to create value for drivers.

Distraction Level API

Addressing developers' misperception of design recommendation in driver interaction chapter of Safe Connectivity Recommendation, Volvo packaged an API for sharing information on driving modes. This so called “distraction level API” afforded developers the opportunity to design applications with different interaction modalities, adapted to particular driving contexts. In other words, the application could use the API to degrade functionality based on the vehicle's status to minimize driver distraction. In using this boundary resource, a development team could focus on other development tasks, while leaving for the Distraction level API to block or degrade functions while driving. A technical report explains:

“This platform resource, considers a fixed amount of distraction levels, which OEMs and developers can freely map to them. In other words, there are five different distraction levels. From the developers' side, the design of the layouts will be changed based on the design guidelines without concerning about the OEM side. OEMs can map their preferred plan based on the chosen regulation. In this solution, developers do not need to be concerned about the OEM's implementation at all.”

Our survey suggested that developers viewed this boundary resources as helpful, with 63% actively involving the API in development and 66.7% considering it as ‘useful’. The in-depth interviews indicated that the API afforded developers to focus on design of functions rather than spending time on safety concerns. As a team member argued; *“we could have a certain interface while it was driving, it reduced the amount of functions available.”* While some teams were unsure about the mechanism of this resource, most developers argued that the API had supported them in designing safe applications, able to adjust interaction modalities for driving situations. One developer noted that using the Distraction Level API *“we could see if the driver was able [allowed] to do some tasks on the app and what functions are not available.”* Another group argued that *“we could use it to make sure that we didn't make some functions on the application available, when it's not appropriate”*. In other words, developers could use this resource to test initial assumptions about an application's functions.

Safety API

To compensate for the weaknesses of the Distraction level API Volvo developed another resource – the Safety API. This platform resource did not represent distraction as fixed levels. In this solution, access was restricted on a functional level rather than on a distraction level. For example, when distraction was high, sound could be disabled. This solution was flexible to the changes from the automaker, and developers did not have to care about distraction levels when coding. Still, developers did not seem to appreciate this boundary resource, with only 14.8% of developers reporting that they actually used it in development. However, this reaction could be explained by developers using other resources instead, such as the Distraction Level API. One developer said that *“the safety API was useless because we got clear instructions by other API [Distraction Level API] on only having two “modes”, standstill or driving.”*

It also appeared as if other boundary resources were more thoroughly introduced (e.g. in introduction lecture), while this API was just briefly described and introduced. One of the developers noted that *“information about the tutorial was insufficient”* argued that the project team *“introduced safety API in very short time at the end of a lecture. We didn't really know what we could do with that.”*

The project team expected it to be hard for external developers to understand this boundary resource and introduced a video tutorial on how to set it up and how to work with it. From Volvo's perspective, this tutorial allowed external developers to get familiar with the Safety API and afforded them requisite skills for how to use it. However, only 18% of the developers used this resource. As they decided not to use the Safety API, they didn't see any value in watching this tutorial. Interviews with developers revealed that developers expected this tutorial to inspire them and suggest affordances, but it focused narrowly on the content and specifications of the API. In summary, this suggests that when developers face a new solution which looks very similar to another existing API, and they are unsure about the affordances of this API, they are not motivated to spend time understanding it or assess its potential value.

Boundary Resources	Definition	Affordance		
		Platform Owner's Perspective	Developers Perspective	
			Developers Perception	Frequency of usage
Safe Connectivity Recommendation	An interpretation of existing recommendations, converted into practical guidelines for how to engage in safe application development on an open automotive platform.	It affords developers to design and develop safe in-car application with relatively limited focus on driver distraction issues.	It affords Self-Assessment/Output Control of Application. It could afford them to fine-tune their application before final demo.	59.3%
Occlusion App 1.0	A development tool, enforcing a particular interaction pattern for use. This app is an augmented application, running in parallel with the application under development.	It affords the capability to explore and experiment with different solutions to fulfill performance test criteria .	It affords Self-Assessment/Output Control of Application. It could afford them to fine-tune their application before final demo.	55.6%
Vehicle Data API	A class of APIs providing access to in-car data, such as Engine statuses, Vehicle Speed and Fuel Level.	It affords easier and preferable access to vehicle signals which in turn, makes it possible to develop tailored applications , providing distinct in-vehicle value.	It affords developing tailored applications providing in-vehicle value . It affords developers to define two modes for the vehicle, which in turn it affords them to restrict some functions according to recommendations.	63%
Distraction Level API	An API providing real-time information on estimated driver distraction in a given situation.	It affords graceful degradation of application functionality, i.e. adapting application requirements on the level and type of interaction to driving context.	It affords testing of developed application's restricted/allowed functions .	63%
Safety API	An API providing real-time information on preferred interaction modalities. In practice, that means recommendations on whether to use particular HMI resources, such as video, images, text, or sound, in a given situation.	Affords external HMI management , i.e. leaving for the vehicle to decide what interaction modalities (video, images, text, sound) to use in a given situation.	Affords testing of developed application's restricted/allowed functions .	14.8%

Table 1. Platform Owner's Perception vs. Developers' perception of Boundary Resources

Discussion

We have explored boundary resource design by applying an affordances perspective. We did so by comparing a platform owners' perceptions of 5 selected platform boundary resources with external developers' perceptions. Our analysis showed that when a platform owner engages in the design of boundary resources they do so by trying to transfer design capabilities through platform resources (Ghazawneh and Henfridsson 2010). These selected boundary resources, designed by Volvo, afforded external developers to design applications for the driving context without distracting the driver.

We studied how boundary resources were designed by Volvo to represent, transform, mobilize and legitimize design knowledge. In designing the safe connectivity recommendation, Volvo provided and shared functional representations, i.e. suggestion for font size and appropriate space between characters in display design recommendation chapter. In addition, some of the resources were designed to transform design knowledge. The Occlusion App 1.0, is an example of such transformation (from a test procedure to a practical tool). External developers used Occlusion App 1.0 and the safe connectivity recommendations to certify, verify and validate the truthfulness and correctness of their design. Using Vehicle data API, afforded developers to discover solutions and create value for users (drivers). We note these affordances align well with previously suggested features of design boundary objects (Bergman et al. 2007).

In some platform resources like Occlusion App 1.0 a combination of features can be seen. However, some developers identified only one of these features. This indicates that even though a platform resource offers

a multitude of different functionalities, a developer might only perceive a selected (limited) amount of them. This potentially prevents the emergence of common representations (Bergman et al. 2007). Our research suggests that platform owners need to work actively and continuously with external developers in order to illustrate the full potential of resources. This might increase the way external developers perceive the functionality, and hence clarify to the platform owners the different affordances (i.e. perceptions of usage) of the resource. Essentially the platform owners may need to tune their resources with developers (Barrett et al. 2012; Eaton et al. 2015)

In an effort to extend boundary resource theory, our research suggests that platform owners' perception of what a boundary resource affords often differ from the external developers' perception, with negative implications on the capability to mobilize for action (Bergman et al. 2007). The Table 1 in the result section illustrates this difference, with the Safe Connectivity Recommendation as a typical example. While the main affordance of this resource was to design and develop an application with limited focus on driver distraction, the resource afforded some developers to control the output of the application. Our case tells us that it is important that platform owners expose boundary resources early, preferably at the same time. This provides developers the freedom of comparing and choosing resource based on their preferences and needs. Further, our study suggests that platform boundary resources should be highly intuitive to support the transfer of design knowledge, described by Bergman et al. (2007). It should be easy for developers to understand what a particular boundary resource affords them, without requiring deeper exploration. They will then perceive it as an assets that enhance, enrich or support application development, not as barrier, requiring exploration. The safety API was perceived as such a barrier, while the Distraction Level API was perceived as simple and direct, making developers use it regardless of its internal complexity. We also note that existing research has connected such "ease of mastery" with generative capacity (Zittrain 2006, p. 1981). Finally, this example elaborates the difference among a firm's logic behind designing a resource and developers' perceptions of the same resource and its affordances (Godoe 2000, p. 1034). Considering different "social worlds" (Star and Griesemer 1989) that developers belong to, their perceptions might be misguided and that there is a need to communicate affordances from the platform owner perspective.

Considering the nature of this study, we identified some limitations in our research. First, in this paper we conducted a single case study and consequently the specific context at Volvo is vital. Besides that, the platform used in this experimental research project was never used in Volvo's ordinary products. We have also considered university students as a sample of third-party developers. We draw our analysis based on their reflection as we believe this was a realistic context to evaluate platform boundary resources. Finally, our research identifies a need to reduce risk and increase precision by rapid, exploratory "pre-tuning" of boundary resources, before introducing them to the final developer community. We leave for future research to explore how such accelerated tuning could be conceptualized.

REFERENCES

- Baldwin, C.Y., and Woodard, C.J. 2008. "The Architecture of Platforms: A Unified View," *Harvard Business School Finance Working Paper:09-034*.
- Barrett, M., Oborn, E., Orlikowski, W.J., and Yates, J. 2012. "Reconfiguring Boundary Relations: Robotic Innovations in Pharmacy Work," *Organization Science* (23:5), pp. 1448-1466.
- Bergman, M., Lyytinen, K., and Mark, G. 2007. "Boundary Objects in Design: An Ecological View of Design Artifacts," *Journal of the Association for Information Systems* (8:11), p. 546.
- Bergvall-Kåreborn, B., Howcroft, D., and Chincholle, D. 2010. "Outsourcing Creative Work: A Study of Mobile Application Development," *ICIS: Citeseer*, p. 23.
- Bosch, J. 2009. "From Software Product Lines to Software Ecosystems," *Proceedings of the 13th international software product line conference: Carnegie Mellon University*, pp. 111-119.
- Boudreau, K.J. 2012. "Let a Thousand Flowers Bloom? An Early Look at Large Numbers of Software App Developers and Patterns of Innovation," *Organization Science* (23:5), pp. 1409-1427.
- Carlile, P.R. 2002. "A Pragmatic View of Knowledge and Boundaries: Boundary Objects in New Product Development," *Organization science* (13:4), pp. 442-455.
- Dougherty, D., and Dunne, D.D. 2011. "Organizing Ecologies of Complex Innovation," *Organization Science* (22:5), pp. 1214-1223.
- Eaton, B., Elaluf-Calderwood, S., Sørensen, C., and Yoo, Y. 2015. "Distributed Tuning of Boundary Resources: The Case of Apple's Os Service System," *ISSUES*.

- El Sawy, O.A., Malhotra, A., Park, Y., and Pavlou, P.A. 2010. "Research Commentary-Seeking the Configurations of Digital Ecodynamics: It Takes Three to Tango," *Information Systems Research* (21:4), pp. 835-848.
- Evans, D.S., Hagi, A., and Schmalensee, R. 2008. *Invisible Engines: How Software Platforms Drive Innovation and Transform Industries*. MIT Press.
- Gal, U., Lyytinen, K., and Yoo, Y. 2008. "The Dynamics of It Boundary Objects, Information Infrastructures, and Organisational Identities: The Introduction of 3d Modelling Technologies into the Architecture, Engineering, and Construction Industry," *European Journal of Information Systems* (17:3), pp. 290-304.
- Gawer, A. 2014. "Bridging Differing Perspectives on Technological Platforms: Toward an Integrative Framework," *Research Policy* (43:7), pp. 1239-1249.
- Gawer, A., Cusumano, M.A., and Strategy, D.S. 2012. "How Companies Become Platform Leaders," *MIT/Sloan Management Review* (49:2).
- Ghazawneh, A. 2012. "Towards a Boundary Resources Theory of Software Platforms,").
- Ghazawneh, A., and Henfridsson, O. 2010. "Governing Third-Party Development through Platform Boundary Resources," *the International Conference on Information Systems (ICIS): AIS Electronic Library (AISeL)*, pp. 1-18.
- Ghazawneh, A., and Henfridsson, O. 2013a. "Balancing Platform Control and External Contribution in Third-Party Development: The Boundary Resources Model," *Information Systems Journal* (23:2), pp. 173-192.
- Ghazawneh, A., and Henfridsson, O. 2013b. "Balancing Platform Control and External Contribution in Third-Party Development: The Boundary Resources Model," *Information Systems Journal* (23:2), pp. 173-192.
- Gibson, J.J. 1986. *The Ecological Approach to Visual Perception*. Psychology Press.
- Godoe, H. 2000. "Innovation Regimes, R&D and Radical Innovations in Telecommunications," *Research Policy* (29:9), pp. 1033-1046.
- Hanseth, O., and Lyytinen, K. 2010. "Design Theory for Dynamic Complexity in Information Infrastructures: The Case of Building Internet," *Journal of Information Technology* (25:1), pp. 1-19.
- Kaplan, B., and Maxwell, J.A. 2005. "Qualitative Research Methods for Evaluating Computer Information Systems," in *Evaluating the Organizational Impact of Healthcare Information Systems*. Springer, pp. 30-55.
- Lee, A.S., and Liebenau, J. 1997. *Information Systems and Qualitative Research*. Springer.
- Leonardi, P.M. 2011. "When Flexible Routines Meet Flexible Technologies: Affordance, Constraint, and the Imbrication of Human and Material Agencies," *MIS quarterly* (35:1), pp. 147-167.
- Levina, N., and Vaast, E. 2005. "The Emergence of Boundary Spanning Competence in Practice: Implications for Implementation and Use of Information Systems," *MIS quarterly*, pp. 335-363.
- Messerschmitt, D.G., and Szyperski, C. 2005. "Software Ecosystem: Understanding an Indispensable Technology and Industry," *MIT Press Books* (1).
- Nambisan, S., and Sawhney, M. 2011. "Orchestration Processes in Network-Centric Innovation: Evidence from the Field," *The Academy of Management Perspectives* (25:3), pp. 40-57.
- Remneland-Wikhamn, B., Ljungberg, J., Bergquist, M., and Kuschel, J. 2011. "Open Innovation, Generativity and the Supplier as Peer: The Case of Iphone and Android," *International Journal of Innovation Management* (15:01), pp. 205-230.
- Simon, H.A. 1996. *The Sciences of the Artificial*. MIT press.
- Star, S.L., and Griesemer, J.R. 1989. "Institutional Ecology,Translations' and Boundary Objects: Amateurs and Professionals in Berkeley's Museum of Vertebrate Zoology, 1907-39," *Social studies of science* (19:3), pp. 387-420.
- Svahn, F., and Henfridsson, O. 2012. "The Dual Regimes of Digital Innovation Management," *System Science (HICSS), 2012 45th Hawaii International Conference on: IEEE*, pp. 3347-3356.
- Tiwana, A., Konsynski, B., and Bush, A.A. 2010. "Research Commentary-Platform Evolution: Coevolution of Platform Architecture, Governance, and Environmental Dynamics," *Information Systems Research* (21:4), pp. 675-687.
- Treem, J.W., and Leonardi, P.M. 2012. "Social Media Use in Organizations: Exploring the Affordances of Visibility, Editability, Persistence, and Association," *Communication yearbook* (36), pp. 143-189.
- Yin, R.K. 2013. *Case Study Research: Design and Methods*. Sage publications.
- Yoo, Y., Henfridsson, O., and Lyytinen, K. 2010. "Research Commentary-the New Organizing Logic of Digital Innovation: An Agenda for Information Systems Research," *Information Systems Research* (21:4), pp. 724-735.
- Zittrain, J.L. 2006. "The Generative Internet," *Harvard Law Review*, pp. 1974-2040.