Author(s): Ojala, Arto; Lyytinen, Kalle

Title: Competition Logics during Digital Platform Evolution

Year: 2018

Version:

Please cite the original version:

Ojala, A., & Lyytinen, K. (2018). Competition Logics during Digital Platform Evolution. In Proceedings of the 51st Hawaii International Conference on System Sciences (HICSS 2018) (pp. 1026-1035). University of Hawai'i at Manoa. Proceedings of the Annual Hawaii International Conference on System Sciences. https://doi.org/10.24251/hicss.2018.130

Competition Logics during Digital Platform Evolution

Arto Ojala
University of Jyväskylä
arto.k.ojala@jyu.fi

Kalle Lyytinen
Case Western Reserve University
kalle.lyytinen@case.edu

Abstract

How are platforms built and how do they evolve? This is a salient question in digital ecosystems, where the competition has moved from traditional one-sided business logics to multi-sided platforms. In this paper, we explore how a digital platform evolves when the organization of the multilayered platform architecture, and related control points, is modified through competitive moves. We also examine how a firm may be able to manage the increased complexity of the platform. We show that when technical and strategic bottlenecks are solved, the platform owner can expand control to strategically important layers of the platform stack. The findings indicate that the complexity of the platform increases through a series of competitive moves. However, complexity can be managed by increasing the standardization of the platform interfaces, and by jockeying for a stronger position in critical parts of the platform stack.

1. Introduction

The digitization of product platforms has spawned services which radically change business models and disrupt ecosystems. The very existence of specific industries has been called into question. For example, LG and Sharp recently launched TV models which feature a software-based game console (client software), and which connect a TV set equipped with a control pad to a gaming platform in the cloud, thereby encroaching on the game console industry. Recent research has significantly increased our knowledge of the organizational and business mechanisms that underlie digital platform innovations (e.g. [1, 2, 3, 33]). They have also shown that the competition results in increased platform complexity (e.g. [4, 5]), and in new competitive landscapes among and within platforms (e.g. [6, 7, 8, 9]). However, we know little of how digital platforms evolve over time, or of the mechanisms that influence such growth – topics that are highly important for platform theory [9, 10]. One key question in understanding digital platform evolution concerns which types of competitive moves[1] constitute a successful evolutionary path for platform growth, given the dynamics of the marketplace.

We also know little of how the design architecture of the digital platform needs to evolve when new competitive moves add, modify, or remove platform features. During this dynamic, a firm typically seeks to create a new market position by modifying elements in one or several layers of the platform stack [3]. The firm can also seek to act as a platform for platforms for services located in other layers of the service stack, and thereby create a new control point. Furthermore, as the overall platform evolves, and as new services and features are added, the number of participating actors will increase, as well as the types of couplings between the actors. This increases the overall complexity of the platform stack and may challenge its effective management. Matters are further complicated by the fact that different actors can now gain access to different control points of the architecture, located in different layers of the platform [5].

Due to the swift changes in the configurations, the control of the overall structure of the platform will change as evolution occurs, especially when specific technologies or services at a certain layer become technical or business-related bottlenecks [11]. These dynamics increase the complexity of the platform, in terms of how the services can be orchestrated and managed. Nevertheless, we currently know little of how overall digital platform management unfolds, due to the fact that studies up to now have primarily investigated the phenomenon statically, over brief time periods [7, 9, 12]. Nor does the extant literature address how the digital platform is built from scratch by a nascent firm, or how the firm then engages in a specific series of competitive moves over time. There are several possibilities to engage in such moves at any given time point, as afforded by the layered architecture. The need is thus to discover specific rules that govern the selection of appropriate moves, given the history that applies, and the specific circumstances. One way to decipher such moves is to track

[1] The aim of a competitive move is to improve, reposition, or defend a firm's position in the market.

URI: http://hdl.handle.net/10125/50017
ISBN: 978-0-9981331-1-9

competitive moves longitudinally within a platform, differentiating moves that center on different layers, and analyzing how such moves relate to the overall logics of the digital platform growth.

Our study aimed to address these gaps in current understandings of platform theory. In particular, we were interested in the kinds of competitive moves that might lead to a software product being "platformized" within chosen markets. Thus, we examined (1) how a digital platform owner was able to reconfigure the platform stack over time by engaging in a specific sequence of competitive moves, (2) to what extent these competitive moves changed elements at different layers of the stack and thereby changed the dynamics of control over the stack, and (3) how such moves may potentially influence platform complexity, including how such complexity can be managed through garnering new firm-level capabilities. In addressing these questions, our study responded to recent calls to investigate factors that may influence a firm's strategic choices in situations of digitally-enabled competition [3], and the dynamics of platform evolution [9, 10, 13]. Our study also sought to shed light on the origins of increasing platform complexity [1, 4, 5]. With these considerations in view, we conducted a longitudinal case study, covering more than a decade, to explore the evolution and competitive logics associated with a digital platform in the gaming industry.

2. Conceptual development

A platform forms a competitive foundation upon which other firms can develop content, innovate with complementary technologies, and create new services [13]. In this paper, we focus on digital platforms, with such a platform defined 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" [10, p. 672]. Such platforms are typically organized as a loosely coupled layered modular architecture. This organization enables firms to innovate simultaneously at different layers of the architecture, and to compete on related services [3]. Most digital platforms operate on two-sided markets; hence, content providers and end-users interact through the platform and thereby enjoy direct and indirect network effects [6, 9]. In such markets, a key challenge is how firms can design, build, control, and sustain the competitiveness of the platform by orchestrating an ecosystem that will invite participation from both sides of the platform [3]. Typically, this happens by positioning the platform through aggressive, novel, and swift competitive moves, aiming to achieve a dominant position on either or both sides of the market.

2.1. Evolution of the platform stack

Digital platforms change by effecting changes at some level of a multilayered platform stack, including the relationships between the stack layers. By tracing such changes, we can understand how the platform is organized, and the essential couplings between different layers [3, 9]. Such connections also reveal how the platform stack is connected to a variety of actors in the overall ecosystem. We can conceptualize the platform stack as a loosely coupled layered modular architecture with four layers, comprising (i) a device, (ii) a network, (ii) a service, and (iv) content. The device layer refers to physical devices that can be used to connect and interact with the platform (e.g. a specific TV set, mobile phone, etc.). The network layer refers to the network protocols that the platform uses to communicate over the network to different devices. The service layer relates to the functionality of the applications that enable customers to use the content-related services. The content layer includes the content that customers interact with, e.g. new videos (cf. [3]).

By studying the platform stack as a layered modular architecture, we can recognize the loosely coupled nature of the digital platform. Loose coupling implies that the platform interacts with an open-ended range of devices, networks, services, and content [3]. Accordingly, the designers of the components of a layer in the layered modular architecture cannot fully predict how the designed component will be used in the future (in the context of a new service), or how it might impact on other layers [3]. Having a layered modular architecture allows innovations to occur independently at any layer. This also means that a change in one layer will propagate non-linear changes in other layers, creating opportunities or restrictions for the platform owner.

There are at least three distinct groups of actors that participate in a digital platform ecosystem organized around layered platform architecture. These are the platform owners, the content providers, and the end-users [9]. Each actor has unique needs and motivations to participate within the different layers of the platform stack. The actors may control different layers of the platform stack, in part or as a whole. The control can be based on either strategic or technical "bottlenecks" [11, 21]. In the event of technical bottlenecks, there may at a certain layer be few or no alternative technologies that a firm can use to bring the platform-related service to the market (cf. [11]). Strategic bottlenecks refer to actions whereby a firm can prevent or limit other actors' access to their platform [11].

Commonly, platform owners will try to attract more content providers for their platform to make it more attractive to platform users [6, 9]. To facilitate such content acquisition, the platform owner can provide content providers with access and resources that will facilitate the development of attractive services for the platform [9, 12]. These resources can be application programming interfaces (APIs) or software development kits (SDKs) that third-party developers can use to bring their innovations more easily to the platform [22].

Because several competing platforms are available on specific layers, platform owners typically seek to "multihome" their services [23, 24]. In multihoming, a content provider may, for example, use more than one channel to offer content to end-users, or end-users can use more than one device or network to gain access to the content [9]. In the context of this study, we are interested in "platform multihoming." In this case, a platform owner seeks to multihome the platform across different channels on the network layer, or across multiple devices in the device layer, so that both service providers and end-users can enjoy more ways to gain access to the service. The approach in our study differed from previous studies on multihoming, in that the latter have primarily focused on multihoming problems from the viewpoint of either the content provider or the end-user (see e.g. [9, 23, 24]).

2.2. Competitive moves

Digital platform evolution consists of a sequence of competitive moves organized at different levels of the multi-layered platform stack. Such competitive moves can be conceptualized as a firm's distinct and discrete actions in the market place [14], with "action" defined as "a specific and detectable market move" initiated by a firm. The moves can either be proactive or reactive. Proactive moves seek to improve or re-position the firm's position in the market, while reactive moves are generated as a response by the firm to defend its position in the market [14]. These moves are informed by a rationale offered by the managers [14], and such a rationale can be expressed in a set of rules anchored in specific cause-and-effect models that are anticipated while engaging in the move [15, 16]. Overall, a firm's strategy can be conceptualized as an emergent pattern of moves and related rationales [14, 15, 16]. In our case, such moves were related to a series of interactions between a platform and its environment. By investigating these moves and related rationales, it was possible to observe and understand the logics of digital platform evolution [10].

Through competitive moves, the firm's platform innovation typically evolves toward a more complex structure. One reason for this is that digital platforms (due to their loosely coupled and networked nature) constitute highly complex structures over time [4, 5]. Moreover, due to the flexibility of the software, such structures can develop and grow in unpredictable ways [9]. One indication of growing complexity is the exponential growth of actors associated with either side of the platform market [8]. Another indication is the growth of new couplings, with different actors on different sides of the platform in different layers. Finally, the diversity and type of connections within the stack also signals increasing complexity (cf. [17]). As the complexity grows, the platform becomes technologically more sophisticated, in connecting to a larger number of components. It will tend to include a larger number of interfaces whereby the size of the codebase grows exponentially. These all increase the internal and external complexity of the platform, and make the platform more challenging to manage and evolve.

When a firm starts to evolve a platform, the strategic goals of the platform may change constantly as new technologies emerge, as market needs shift, and as consumer or technology preferences change (cf. [8, 18]. To better understand the competitive logic of platform creation, one can adopt some ideas from complexity theory [19], especially regarding nonlinear outcomes based on rapid transitions and co-evolutionary processes, and the ways in which order emerges from such processes [18, 20] In the present study, we were especially interested in how a digital platform owner can create order by stabilizing and destabilizing the platform stack in non-linear ways, through a sequence of competitive moves.

3. Research method

To address our research questions, we conducted a longitudinal, exploratory case study. We chose this method because it made it possible to cater for the empirically rich and detailed data of a complex and understudied phenomenon [25, 26]. The longitudinal case study method also facilitated an examination of long-term changes in the case firm's competitive landscape. We selected the case firm for this study using three criteria: (i) the firm develops digital platform for multi-sided markets, (ii) the firm and the platform have a long history, making it possible to observe the competitive moves over an extended stretch of time, and (iii) the firm is relatively small, making it easier to observe actions taken in the development process within the market [27].

3.1. The case firm

The case firm, G-cluster, has developed a digital platform for cloud gaming services. The platform allows computer and console games to be played across various devices. The platform is operated by the game servers that transmit games as (i) an MPEG stream to client devices over a network operator's landline or high-speed wireless network, or (ii) over-the-top (OTT), as a stand-alone game service. The client devices receive the stream, display the game, and transmit the users' commands back to the game server operating the game. G-cluster was founded in Finland in 2000. Currently, G-cluster has its headquarters in Japan, and its R&D activities in Finland.

3.2. Data collection

We collected several types of empirical material covering the entire history of the case firm, from 2000 to 2015. The most important source was interviews with the managers of the firm. These took place between 2005 and 2016. In total we conducted 28 interviews, each lasting around 45–90 minutes. Because the case firm is relatively small, interviews with the Chief Executive Officer (CEO) formed the main source of information. However, to improve the validity of the study, to avoid personal bias, to triangulate, and to gain the most relevant knowledge for each topic covered [29], we interviewed eight additional employees in the firm. To reduce recall bias, we also interviewed the CEO and four employees more than once. The interviewees were selected on the basis of their knowledge related to various phases of the platform evolution and market development. Furthermore, we conducted interviews with three employees of the case firm's main partner in Japan to acquire a more comprehensive understanding of the evolution of the platform.

All the interviews began with background information on the interviewee, including the interviewee's role in the firm, and involvement in platform development. The first interviews focused on the history of the firm with respect to the creation of the platform and its initial development. Thereafter, each follow-up interview focused on platform and business development since the previous interview. All the interviews were recorded and transcribed verbatim. This resulted in 305 single-spaced pages of interview data. To avoid retrospective bias [29], we collected around 180 pages touching on several types of secondary data. These covered the whole history of the firm, the aim being to validate the interview data whenever possible.

3.3. Data analysis

Inductive techniques were applied to analyze the qualitative data acquired from the case firm [27, 28, 30]. We first carried out data reduction [28] by synthesizing the complete transcripts from the interviews and secondary data [27] into a baseline document covering the history of the firm. Here we followed Pettigrew [31], who recommends arranging incoherent aspects of context evolution in chronological order, the aim being to facilitate understanding of the causal links between critical events.

After the data reduction, we coded the interview data using open thematic content analysis [32]. As a first step, we traced the emerging stages in the platform evolution, using the framework for layered modular architecture devised by Yoo et al. [3] as a template for coding. On the basis of the framework, we coded for all the changes in the different layers of the stack for each competitive move. We next organized these changes into a sequence of "platform stacks" arranged in chronological order, seeking thus to synthesize the entire history of platform configurations. Thereafter, we attempted to discover and identify the motivations for the competitive moves, with reference to the case history. Here our approach was similar to that of Woodard et al. [2]. In applying this method, a move was conceptualized as a logical grouping of sequential changes in the platform stack.

4. Findings

4.1. Competitive moves and changes in the platform stack

Figure 1 gives an overview of the changes in the platform stack, depicting the relevant actors at each layer and the technology/service provided at each stage. The colored boxes illustrate changes associated with each move when new technology, or a new service, was added to some layer. Table 1 synthesizes each competitive move and its scope within the platform stack; it also gives a brief description of the success of the move, and initial reasons behind the competitive move. By reviewing changes in the platform stack we were able to trace how the evolution of the platform and related stack configurations impacted on the growth of the multisided market.

4.1.1. Platform stack #1. The first two competitive moves were associated with initial development and experimentation pertaining to the digital platform for true commercial use. At its inception, G-cluster focused on finding a device that was "right" in terms of bringing to the market a general, mobile gaming service. Accordingly, the first platform stack, configured in 2000–2001, was created by two consecutive competitive moves. The first competitive move was the launching of the gaming service for 3G networks and the introduction of client software for 3G mobile phones. However, at this point there were no operational and robust 3G networks available; hence the market penetration of the 3G technologies proved to be much slower than initially predicted. This meant that the first move failed because of the technical limitations of the network layer.

As their second competitive move, G-cluster piloted a service using a server that was connected to PDAs over the Wi-Fi network. The pilot worked and they were able to show the value of the platform to both sides of the market (value proposition). Though the pilot was successful, the development of the supporting infrastructure (3G networks and 3G mobile phones) and markets was much slower than had been initially estimated. In addition, 3G networks proved unreliable for handling the real-time bit-stream without latency. Hence, G-cluster fairly quickly abandoned the idea of 3G networks, moving instead to offering games on fixed networks.

4.1.2. Platform stack #2. The third competitive move was based on G-cluster's realization that it had to configure its platform stack for fixed networks. The reconfiguration adopted in 2002 was significant, including as it did changes to the platform stack at the device, network, and content layers. The aim was to home the gaming service into the IPTV markets that were emerging as a new way to deliver digital content to televisions using set-top boxes. The solution was successfully implemented on a pilot basis. However, the IPTV operator headed into financial problems, and was unable to continue cooperation with G-cluster.

4.1.3. Platform stack #3. The third platform stack configuration, in 2003–2004, encompassed three competitive moves. The platform, with its capabilities to deliver real-time game content between terminal devices and a server, attracted increasing interest in the marketplace. As a result, G-cluster was acquired by a Japanese firm, Broadmedia (competitive move #4). At the time, Broadmedia was a part of a large Japanese telecom corporation (SoftBank) that owned Japan's largest ADSL network. The firm was actively developing IPTV services, including video-on-demand services, within Japan. This move helped in gaining access to the Japanese markets and in reaching towards an existing customer base. In addition, it facilitated the acquisition of new and more advanced content from game publishers at the content layer.

Though G-cluster had its platform configured for IPTV providers, the integration of the client software to the set-up boxes used for IPTV services proved to be more difficult than expected (competitive move #5). The difficulties were related again to the immaturity of IPTV services. The corporation had successfully launched its video-on-demand services, but it did not want to risk this service by integrating it with a new service offered by G-cluster. In addition, only a limited proportion of consumers used IPTV services when the technology was made available. Altogether, commercialization of the platform was unsuccessful at this stage, because of the immaturity of the relevant network infrastructures, markets, and supporting ecosystems. Because the commercialization of the service to IPTV networks failed, G-cluster configured the platform stack for existing Internet technology as their next competitive move, #6. Device and network layers were reconfigured for PC and Mac computers, and at the service layer, client software was developed for PC and Mac. In practice, players downloaded client software for their computers from the Yahoo BB Japan website. Thereafter, they were able to play casual games over the broadband connection. Altogether, the third platform stack reconfiguration made it possible to commercialize the service.

4.1.4. Platform stack #4. The fourth platform reconfiguration made it possible to expand the device layer to set-top boxes via IPTV operators. Though the previous competitive move had made it possible to commercialize the service, the PC and Mac markets proved unprofitable, because of intense competition from the traditional computer game market. Hence, to expand the market potential for the platform, the company actively searched for potential IPTV providers, having already configured client software for set-top boxes. Accordingly, in 2005 G-cluster got its first IPTV customer, the Cyprus telecommunication Authority (CYTA). This competitive move, #7, enabled the company to expand tits device portfolio from PC and Mac users to IPTV users, and to "multihome" its gaming service.

4.1.5. Platform stack #5. In 2008, as its next competitive move, #8, G-cluster started to configure its platform stack for cable TV networks in the USA. Cable TV operators have traditionally had a strong position in the USA market, and the market commands a huge customer base. The strong position of the cable TV operators in the market meant that the number of IPTV providers (through the ADSL network) was small. For these reasons, G-cluster began to configure its platform and client software on Data Over Cable Service Interface Specification (DOCSIS) networks, and on related DOCSIS set-top boxes. However, the configuration proved to be difficult to implement, because in most cases, the return channel from the end-users to the server running the game was too slow. As a result, G-cluster was not able to commercialize its service in the DOCSIS network.

4.1.6. Platform stack #6. The next platform stack configuration, resulted from two competitive moves. Earlier competitive moves had brought reliability to the platform service and demonstrated proofs of the concepts. This enabled the company to start negotiating with large telecom operators who could offer similar gaming services on their networks. Competitive move #9 was related to the launch of the service in France. G-cluster started to cooperate with a French telecommunication company (SFR), and in 2010 jointly commercialized the service in France through SFR's IPTV network.

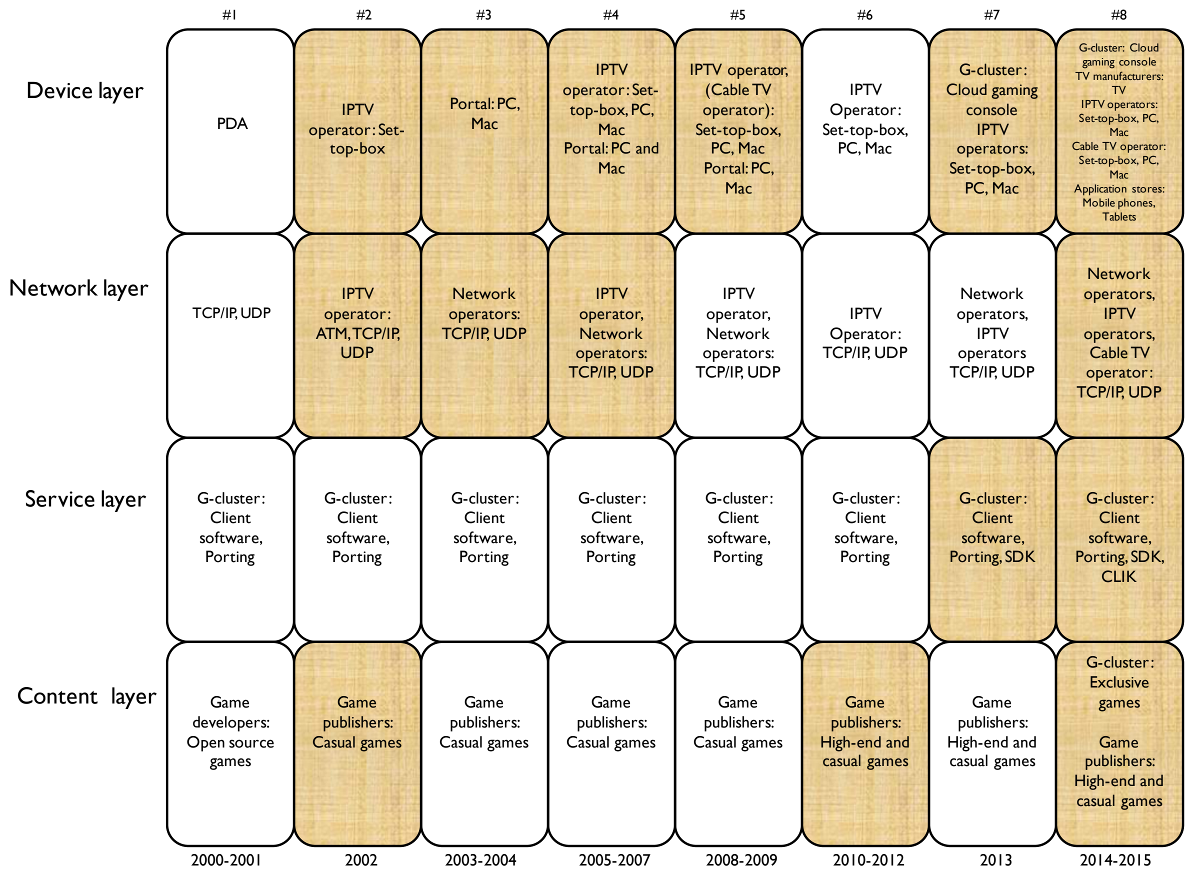

Figure 1. Platform stack configurations

Platform stack	Competitive move	Outcome of the competitive move	Reason for the competitive move
Platform stack #1 20002001	#1: Service for mobile phones to enable stream mobile games over 3G network #2: Client software for PDAs to pilot how the service works	#1: Unsuccessful (Technical bottleneck) #2: Successful	#1: Market entry #2: Value proposition
Platform stack #2 2002	#3: Client software for set-top boxes	#3: Unsuccessful (Technical bottleneck)	#3: Market entry
Platform stack #3 2003-2004	#4: Acquisition by Broadmedia #5: Client software for set-top boxes #6: Client software for PC and Mac	#4: Successful #5: Unsuccessful (Technical bottleneck) #6: Successful	#4: Market entry #5: Market entry #6: Market expansion/Value proposition
Platform stack #4 2005-2007	#7: Client software for set-top boxes	#7: Successful	#7: Market expansion/ Value proposition
Platform stack #5 2008-2009	#8: Client software for cable TV set-top boxes	#8: Unsuccessful (Technical bottleneck)	#8: Market expansion
Platform stack #6 2010-2012	#9: Platform commercialized by a reliable partner #10: SDK for game publishers	#9: Successful #10: Successful	#9: Market expansion/Value proposition #10: Value proposition
Platform stack #7 2013	#11: Development of cloud gaming console	#11: Successful	#11: Market expansion/Value proposition
Platform stack #8 2014-2015	#12: Client software for TVs #13: CLIK for game developers #14: Client software for 4G mobile phones. #15: Exclusive game content for the platform #16: Client software for cable TV set-top boxes	#12: Successful #13: Successful #14: Successful #15: Successful #16: Successful	#12: Market expansion/Value proposition #13: Value proposition #14: Market expansion/Value proposition #15: Value proposition #16: Market expansion/Value proposition

Table 1. Competitive moves related to each platform stack configuration

This increased the credibility of the service with other telecom operators. Following its successful launch of the service, in 2012, G-cluster gained its next content delivery contract with Orange, also for the French market. The increased size of the market helped it to negotiate and acquire advanced and recent content from game publishers. The service started to attract increased attention among well-known game publishers, such as Disney, Warner Brothers, and Electronic Arts, all of which started to offer content on the platform. These contracts enabled the company to expand the content layer from casual games to the latest high-end games.

In 2010, G-cluster launched its own SDK for game publishers, as competitive move #10. The SDK facilitated the porting of games to the platform. Now the game developers were able to write their games directly to G-cluster's platform.

Altogether, this platform stack reconfiguration introduced important changes, enabling initial stabilization of G-cluster's position in the market. In the first place, it decreased complexity, as the service was now provided purely via an IPTV operator that controlled the device and network layers. Secondly, it permitted duplication of the service for other IPTV providers, without increasing the service complexity. Thirdly, it attracted the attention of content providers; this helped in expanding the actors within the content layer, and in attracting potential IPTV partners.

4.1.7. Platform stack #7. Although the business through the IPTV operators was steadily growing, and although G-cluster received new content for its platform from several well-known game publishers, the delivery of the game content was limited to the IPTV operators. This made the total market coverage for the games relatively small. Furthermore, G-cluster realized that the negotiation processes for configuring client software for the operators' set-top boxes was slow and time consuming. These difficulties led to competitive move #11. The aim was to expand the market and avoid the difficulties associated with IPTV operators, by-passing the strategic bottleneck that the latter created. This was accomplished by developing a cloud-based game console. In 2013, G-cluster launched a cloud game console, called the "G-cluster gaming machine" in Japan, and later in the USA. The console was a small device with HDMI/USB adapters that enabled end-users to gain access to G-cluster's cloud game server. As the cloud console used the OTT network, the service was available through any telecom operator's broadband network. The cloud-based gaming console thus made it possible to expand market control at the device layer, i.e. beyond IPTV operators.

4.1.8. Platform stack #8. In 2013 G-cluster decided to integrate its client software directly into (digital) TVs as their next competitive move (#12). This further expanded multihoming at the device layer. Bringing the cloud game console to the market also convinced TV manufacturers that the service would work over an OTT network. In late 2013, G-cluster signed a contract with two TV manufacturers (Sharp and LG), both of which integrated G-cluster's client software for their initial TV setup. This was brought to the Japanese market in 2014.

Because of the increased number of content delivery channels, G-cluster needed more content for their service. To speed up the acquisition and integration of the gaming content for its platform, G-cluster developed a Codeless Integration Kit (CLIK) as its competitive move #13. CLIK demonstrated to game publishers how their games would work in the cloud environment, reduce the need for porting, and provide a quicker way to acquire content for the platform.

In late 2014, G-cluster finally started to develop the platform toward its original target device, i.e. mobile phones (competitive move #14). For this purpose, the company established cooperation with Square Enix, which develops the role-play series *Final Fantasy*. Games that require about 60 gigabytes of memory to install, and which cannot be run on mobile devices, can now be played on mobile devices over 4G or Wi-Fi as a cloud service. G-cluster has configured the client software for the mobile games, and the client has been made available in Japan through Apple's App Store and Google's Play store.

In 2014 G-cluster expanded its activities to the content layer by starting to develop exclusive content for its platform, seeking thus to attract more customers (competitive move #15). The games in question included features that permitted, for example, offline gaming using a mobile phone, and thereafter transference of the game figure from the mobile phone to TV. Continuation was possible by playing the game in the cloud environment, with more advanced graphics and features. In 2015, the company also expanded its market by developing client software for a Japanese cable TV provider J:COM (competitive move #16). The development of cable TV networks made it possible to run the service over the cable network.

4.2. Competitive moves and the reconfiguration of the platform stack over time

As can be observed from Figure 1, most of the platform stack reconfigurations prior to 2010 took place at the network and device layers, whereas later reconfigurations increasingly involved changes at the service and content layers. This marks a shift from removing technical bottlenecks and related control points towards creating new value propositions in growing markets, and either creating or overcoming strategic bottlenecks. In specific terms, the first competitive moves were related to addressing an initial homing problem, i.e. finding a suitable network/device combination that would be sufficient to bring the service to the market. Competitive moves #1–#5 focused mainly on overcoming technical bottlenecks related to low bandwidth at the network layer, and to the unreliable set-top box technology at the device layer. Platform stack reconfiguration #3 made it possible to commercialize the service through PCs and Macs, and reconfiguration #4 – operating at the network and device layers – expanded the service for IPTV. However, technical bottlenecks were also present at platform stack reconfiguration #5, when G-cluster tried to enter IPTV markets through cable-TV operators. Here one can see how the instability of the technologies at the network and device layers hindered commercialization of the service in the early stages of platform evolution. Finally, competitive move #9 led to platform stack reconfiguration #6. This made it possible to commercialize the service through a credible and well-known IPTV provider.

After platform stack reconfiguration #6, the competitive moves focused on market expansion and on creating novel value propositions on either side of the market. This included the introduction of new network and device combinations, managed via the same cloud-based solution. Considered in detail, market expansion was motivated by competitive moves #11, #12, and #16, within a device layer that was aimed to increase the multihoming of the platform via several devices. This increased the value of the platform to the content providers and end-users. Competitive moves #10 and #13-#15 (which were related to value propositions) focused mainly on the service and content layers. They facilitated new and more advanced content acquisition via SDK and CLIK at the service layer.

4.3. The dynamics of the control over different layers of the platform stack

For the most part, G-cluster operated within and sought to control the service layer. However, the actors in the network layer controlled access to the device layer during all of platform stack reconfigurations #1–#6. These actors created both technical and strategic bottlenecks, since they controlled access and constrained possibilities to multihome the service at the device layer. Initially, technical bottlenecks played a major role, leading to constant platform stack reconfigurations in attempts to find a suitable combination at the network and device layers, with a view to creating a feasible service. Thereafter, the control points moved toward strategic control, within which IPTV providers controlled access to the devices and to end-users. Competitive move #11 and platform reconfiguration #7 made it possible to bypass this control point. The competitive moves here created autonomy through use of the company's own device (a cloud gaming console); this provided direct access to the device layer via an operator-independent OTT model. After competitive move #11, G-cluster was in part able to take control of the device layer.

On the other side of the platform, the content providers exercised strong control over the content provided by the platform. During platform stack reconfigurations #1–#5, the content providers were cautious about releasing content. This was mainly because of the undeveloped device and network technologies, plus relatively small market coverage. However, competitive move #9 opened up a larger market through a well-established IPTV provider, increasing the content provider's willingness to provide content. In addition, G-cluster expanded its activities towards the content layer when it started to develop exclusive content for its service (competitive move #15).

4.4. Managing increasing complexity through competitive moves

The complexity of the platform increased through competitive moves #1–#8, as the platform had to be individually adjusted to different networks and devices. These increased the couplings between layers and related actors. As the complexity of the platform increased, it became increasingly difficult to manage the platform adequately. To solve the problem, G-cluster implemented competitive move #9. This reconfigured the platform stack in such a way that its complexity decreased substantially. In this stack configuration, G-cluster created a new order by simplifying its network and device layers to such an extent that the company operated through only one reliable actor in the market. Moreover, the complexity could now be managed at the technical level through the increased technical maturity of the platform (competitive move #9). It was now possible to standardize interfaces with (i) the invoicing system, (ii) terminal devices, and (iii) channel menu selection. There was now a situation involving fewer actors and greater standardization of the interfaces.

Platform complexity started to increase once again after platform stack reconfiguration #6. This involved a series of competitive moves: #11, #12, #14, and #16. These moves increased the number of actors and technologies involved at the network and device layers. In addition, the number of content providers increased at the content layer. However, by this time, G-cluster had built up its capabilities to manage the increased complexity. The measures used included (i) creating a platform with standardized interfaces, (ii) controlling installations that had previously been coordinated through third parties, (iii) harmonizing the licensing model, and (iv) standardizing content acquisition via SDK and CLIK.

5. Conclusions and discussion

This exploratory case study makes three notable contributions to previous literature and theory development in the context of digital platform evolution. Thus, our findings reveal: (1) how a digital platform evolves through competitive moves, (2) how competitive moves change the dynamics of control of the platform stack, and (3) how a platform owner can manage the increasing complexity.

In the first place, the study illustrates how the platform owner can reconfigure the platform stack [3] over time by engaging in a sequence of competitive moves. In other words, by studying a platform change longitudinally, one can see how a nascent firm is able to build up a platform from scratch through an emergent pattern of competitive moves (cf. [14, 15, 16]), and further, how the platform evolves over the period in question. The study contributes to competitive dynamics theory [14] in the context of digital platform evolution. The findings indicate that the possibility to reconfigure the platform stack is largely dependent on previous competitive moves, and on the capability of overcoming technical and strategic bottlenecks [cf. 11, 21]. This expands from competitive dynamic literature (see Chen and Miller [14] for a further review), which has ended to focus on the market entry of new products or services in traditional industries. In contrast, the present study shows how the same service can be applied to create new markets and value in multisided markets. Our conceptualization indicates that initial moves aim to multihome the platform and to remove technical bottlenecks. Thereafter, competitive moves focus on service and content layers, with the aim of creating value to end-users and content providers within a chosen niche. As observed here, the advances of network and device technologies may also largely dictate the rationality of competitive moves.

Secondly, our findings contribute to an understanding of how competitive moves change the dynamics of control over the layers of the platform stack. It can be concluded that within sequences of competitive moves, technical bottlenecks arise; these needs be solved at critical layers before a firm can consider expanding its control over other strategically important layers. In other words, by overcoming technical bottlenecks [11], a firm creates value and capabilities for subsequent moves aimed at expanding control over the platform stack. The overcoming of bottlenecks requires high scalability of the service to new devices and networks. Increased control also facilitates the outsourcing of value creation activities to other actors within the platform stack, as the actors become increasingly dependent on the platform owner. In the present case, the platform owner initially operated solely in the service layer. However, once technical bottlenecks were solved, the platform owner started to expand control over other strategically important layers (device and content). The rationality of expanding control was related to the need to overcome the strategic bottlenecks set by IPTV providers at the device and network layers, and to create more differentiated value for end users. Consequently, this increased the value of the platform for both sides of the market (cf. [3, 22]). The findings in this respect advance digital platform literature [1, 3, 5, 12] by demonstrating how a digital platform owner may be able to shift control from one layer to other layers.

Thirdly, we contribute to complexity theory [18, 19, 20] in the context of digital platform evolution, by examining how competitive moves increase the complexity associated with a platform, and how such complexity could be managed. We indicate that platform complexity increases during an evolutionary process within which a firm overcomes technical or strategic bottlenecks, and moves to new markets by multihoming the platform. However, the complexity can be managed through competitive moves that focus on simplification, standardization, and control of the platform stack. By reducing the complexity, a platform owner can gain a more competitive position in the market, through having better control over different layers of the platform stack.

Taking a broader view, the findings extend platform complexity research [4, 5] by demonstrating the reasons for increased platform complexity, and further, how the complexity can be managed through specific competitive moves. In particular, the findings indicate that the complexity of the platform increased during successive competitive moves, as new technologies emerged (cf. [18]), making it possible to multihome the platform across different devices and networks. In the present case, this increased the number of actors and couplings, not just on both sides of the platform [6, 8, 9], but also within different layers of the stack, making the management of the platform increasingly complex. This illustrates how a change at one layer of the platform stack can impact on other layers in the stack, thus demonstrating the nonlinear outcomes of rapid co-evolution that are common for complex systems [18, 19]. All in all, it can be claimed that the findings lead to a better understanding of how firms may be able to manage and decrease platform complexity.

6. References

[1] Tilson, D., Lyytinen, K., and Sorensen, C. 2010. Digital Infrastructures: The Missing IS Research Agenda, *Information Systems Research* 21:4, 748-759.

[2] Woodard, C.J., Ramasubbu, N., Tschang, F.T. and Sambamurthy, V. 2013. Design Capital and Design Moves: The Logic of Digital Business Strategy, *MIS Quarterly* 37:2, 537-564.

[3] Yoo, Y., Henfridsson, O., and Lyytinen, K. 2010. The New Organizing Logic of Digital Innovation: An Agenda for Information Systems Research, *Information Systems Research* 21:4, 724-735.

[4] Tilson, D., Sorensen, C., and Lyytinen, K. 2012. Platform Complexity: Lessons from Mobile Wireless, *International Conference on Mobile Business*, Paper 6.

[5] Tilson, D., Sorensen, C., and Lyytinen, K. 2013. Platform Complexity: Lessons from the Music Industry, *46th Hawaii International Conference on System Sciences (HICSS)*, 4625-4634.

[6] Eisenmann, T., Parker, G., and Van Alstyne, M.W. 2006. Strategies for two-sided markets, *Harvard business review* 84:10, 92-101.

[7] Gawer, A., and Cusumano, M.A. 2008. How Companies Become Platform Leaders, *MIT Sloan Management Review* 49:2, 28-35.

[8] Parker, G.G., Alstyne, M.W.V., and Choudary, S.P. 2016. *Platform Revolution.* W.W. Norton & Company.

[9] Tiwana, A. 2013. *Platform Ecosystems: Aligning Architecture, Governance, and Strategy*, Amsterdam: Waltham, MA, Morgan Kaufmann.

[10] Tiwana, A. Konsynski, B. and Bush, A.A. 2010. Platform Evolution: Coevolution of Platform Architecture, Governance, and Environmental Dynamics, *Information Systems Research* 21:4, 675-687.

[11] Baldwin, C.Y. 2015. Bottlenecks, Modules and Dynamic Architectural Capabilities, *Harvard Business School Finance Working Paper* 15-028.

[12] Eaton, B. 2016. The Dynamics of Digital Platform Innovation: Apple's Strategy to Control Modular and Architectural innovation in OS, *49th Hawaii International Conference on System Sciences*, 1287-1296.

[13] Gawer, A. 2009. Platform dynamics and strategies from product to services, in *Platforms, Markets and Innovation*, (A. Gawer ed.), Edward Elgar Publishing Inc, 45-76.

[14] Chen, M-J., and Miller, D. 2012. Competitive Dynamics: Themes, Trends, and Prospective Research Platform, *Academy of Management Annals* 6. 135–210.

[15] Baum, J.A.C., and Korn, H.J. 1996. Competitive Dynamics of Interfirm Rivalry, *Academy of Management Journal* 39:2, 255-291.

[16] Mintzberg, H. 1978. Patterns in Strategy Formation, *Management Science* 24:9, 934-948.

[17] 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, 1-19.

[18] Benbya, H., and McKelvey, B. 2006a. Toward a complexity theory of information systems development, *Information Technology & People* 19:1, 1234.

[19] Holland, J.H. 1995. *Hidden order: How adaptation builds complexity*. Basic Books.

[20] Mainzer, K. 1994. *Thinking in Complexity: The Computational Dynamics of Matter, Mind, and Mankind*, Springer Science & Business Media.

[21] Jacobides, M.G., and Tae, C.J. 2015. Kingpins, Bottlenecks, and Value Dynamics Along a Sector, *Organization Science* 26:3, 889-907.

[22] Yoo. Y., Boland, R.J., Lyytinen, K., and Majchrzak, A. 2012. Organizing for Innovation in the Digitized World, *Organization Science* 23:5, 1398-1408.

[23] Hyrynsalmi, S., Suominen, A., and Mäntymäki, M. 2016. The influence of developer multi-homing on competition between software ecosystems, *Journal of Systems and Software* 111, 119-127.

[24] Landsman, V., and Stremersch, S. 2011. Multihoming in Two-Sided Markets: An Empirical Inquiry in the Video Game Console Industry, *Journal of Marketing* 75:6, 39-54.

[25] Edmondson, A.C., and McManus, S.E. 2007 Methodological fit in management field research, *Academy of Management Review* 32:4, 1155-1179.

[26] Yin, R.K. 2009. *Case study research: Design and methods*. CA: SAGE Publications.

[27] Eisenhardt, K.M. 1989. Building Theories from Case Study Research, *Academy of Management Review* 14:4, 532-550.

[28] Miles, M.B., and Huberman, A.M. 1994. *Qualitative Data Analysis: An Expanded Sourcebook*, Sage Publications.

[29] Huber, G.P., and Power, D.J. 1985. Retrospective Reports of Strategic-level Managers: Guidelines for Increasing their Accuracy, *Strategic Management Journal* 6, 171-180.

[30] Eisenhardt, K.M., and Graebner, M.E. 2007. Theory building from cases: Opportunities and challenges, *Academy of Management Journal* 50:1, 25-32.

[31] Pettigrew, A.M. 1990. Longitudinal Field Research on Change: Theory and Practice, *Organization Science* 1:3, 267-292.

[32] Corbin, J., and Strauss, A. 2014. *Basics of qualitative research: Techniques and procedures for developing grounded theory,* Sage publications.

[33] Ojala, A. 2016. Business models and opportunity creation: How IT entrepreneurs create and develop business models under uncertainty. *Information Systems Journal* 26:5, 451-476.