Association for Information Systems
AIS Electronic Library (AISeL)

MCIS 2018 Proceedings

Mediterranean Conference on Information Systems (MCIS)

2018

Managing Organizatinal Resources as Platform Boundary Resources

Amir Mohagheghzadeh
University of Gothenburg, Gothenburg, Sweden, amir.mohagheghzadeh@ait.gu.se

Juho Lindman
University of Gothenburg, Gothenburg, Sweden, juho.lindman@ait.gu.se

Jennifer Horkoff
University of Gothenburg, Sweden, jenho@chalmers.se

MANAGING ORGANIZATIONAL RESOURCES AS PLATFORM BOUNDARY RESOURCES

Research full-length paper

General Track

Mohagheghzadeh, Amir, University of Gothenburg, Gothenburg, Sweden, amir.mohagheghzadeh@ait.gu.se

Lindman, Juho, University of Gothenburg, Gothenburg, Sweden, juho.lindman@ait.gu.se

Horkoff, Jennifer, Chalmers and University of Gothenburg, Gothenburg, Sweden, jenho@chalmers.se

Abstract

Approaching digital innovation via digital platforms shifts firms' locus of attention to the different actors in their ecosystems. Firms tend to empower the platform's ecosystem through expanding developer contribution via introducing boundary resources such as application programming interfaces (APIs). This study addresses the challenge of platform owners managing their internal assets as platform boundary resources. We seek to answer how platform owners can identify and visualize values of potential boundary resources by conducting a single case study at a large international company active in embeded software development area. This study suggests e3 value modelling as a tool to assist platform owners in understanding the platform ecosystem actors, the values of assets for the ecosystem and how these values can be interchanged among the actors.

Keywords: Application Programming Interfaces, Digital Platform, Boundary Resources, Value Modelling.

1 Introduction

Digitalization has revolutionized industries and industrial practices, but the impacts of digitalization can be studied from different perspectives. Digital platforms draw incumbent firms' attention to the ecosystem and the potential value of the participants in the ecosystem. Using digital platforms, without any considerable marginal costs, firms can benefit from reproduction of applications (Benkler, 2006; Brynjolfsson & Saunders, 2009; Shapiro & Varian, 1999). Third-party developers may play a significant role in the growth of the ecosystem via developing a wide range of innovative applications (Yoo, Henfridsson et al. 2010). Given the crucial role of third-party developers, platform owners tend to support the developers and fulfil their demands to facilitate their contribution to the ecosystem. The firms' business models can be adapted to external actors' contributions.

Existing literature suggests the concept of platform boundary resources to shift design capability to developers, to facilitate use of a platform's core functionality and to develop third-party applications (Ghazawneh & Henfridsson, 2013; Prügl & Schreier, 2006; Svahn, Lindgren, & Mathiassen, 2015) on one hand and govern the developers' output on the other hand. Platform owners can also establish arms-length relationships with developers through these resources. These resources can be in the form of SDKs (software development kits), APIs (application programming interfaces), incentives, IP rights, guidelines or documentation (Ghazawneh and Henfridsson 2013).

Research shows that platform owner can use boundary resources to govern development practices. In addition, there are research studies on designing and reshaping platform boundary resources(Eaton, Elaluf-Calderwood, Sorensen, & Yoo, 2015; Mohagheghzadeh & Rudmark, 2017), and transformation of internal resources into platform boundary resources (Mohagheghzadeh & Svahn, 2016a, 2016b).

Exposing private company assets such as internal APIs strategically via boundary resources (to third-party development) may lead to unexpected fertility in innovations. On the other hand, this action may result in a number of risks: losing the organization's competitive advantage in the market, security risks or privacy issues. In order to facilitate strategic APIs management, platform owners need to have a visual and conceptual understanding on the value of the assets for the ecosystem.

One of the challenge on this journey is to make the right decision on whether to expose or keep an asset internally. A practical tool that facilitate decision making in different situations such as development of new API or evaluating an existing API is needed. In this study, by contributing to the boundary resource theory, we are investigating the platform owner's approach in finding the suitable framework to visualize and assess the APIs.

In this effort, we tried to answer the question: *"How can platform owners identify and visualize values of potential boundary resources?"*

In what follows, we will review the related studies on boundary resources and their management. Then, we will explain the chosen research approach. After this, the findings of this study are presented, and finally, we will review the findings to answer the research question in the discussion section.

2 Theoretical Background

2.1 Internal APIs

To conduct this study, we departed from the resource-based view (RBV) of the firm. The original argument of RBV is that competitive advantage mainly lies in the application of a package of precious tangible or intangible assets at the firm's disposal (Barney, 1991; Penrose, 2009; Wernerfelt, 1984). However, when seeking to transform existing organizational and technological resources, firms need to pay careful attention to what they actually afford. As Edith Penrose (1959) suggested, "it is never resources themselves that are the 'inputs' in the production process, but only the services that the resources can render" (Penrose, 2009, p.22). While a specific resource can provide particular affordance within the organization, introducing the same resource to the third-party developers may deliver other design capabilities to the external actors in the ecosystem. On the other hand, the new resource in the ecosystem may result in new affordance and competence in the market. Consequently, a decision to expose a resource to the external actors in the ecosystem might result in a number of contradictions and conflicts of interest.

API development and deployment allow downstream developers to access assets such as data, devices and services. Platform owners decide whether to primarily use APIs internally or to expose them to third parties. This decision of what to expose or not is increasingly seen as a strategic one. API providers generally monitor and control access to the exposed asset to some degree. The benefits of APIs are related to layering common to digital technologies (Yoo, Henfridsson, & Lyytinen, 2010): The API selectively reveals functionalities, making it easier to access the exposed asset (de Souza & Redmiles, 2009). Additionally, this setup allows for novel mechanisms of stigmergic coordination (Bolici, Howison, & Crowston, 2016). Several industrial practice reports offer design considerations for APIs. For example, Apigee1 provides information on API usage monitoring, monetization and exposure

---

[1] https://apigee.com

management of open APIs. However, research on the management of APIs is still an area of rapid scientific development, and there are many questions related to the organizational impacts of APIs and intertwined business decisions.

2.2 Platform Boundary Resources

A number of researchers have investigated third-party development in relation to digital platforms by focusing on topics such as incentives of developers (Tiwana, Konsynski, & Bush, 2010), development practices (Tiwana et al., 2010), generativity (Evans, Hagiu, & Schmalensee, 2008) and value of their contributions (Messerschmitt & Szyperski, 2005). To provide a wide variety of applications to the end users, the platform owner should support independent development. Digital platforms are subject to indirect network effects. By inviting more developers to the ecosystem, more applications can be developed (Boudreau, 2012). More applications can satisfy a wider range of end users, which in turn results in an extended installed base. This will motivate more third-party developers to join the platform and offer new applications. The virtual cycle of value creation and capture described here is based on indirect network effects. Thus, facilitating third-party development is one of the highest priorities of the platform owners.

A boundary resource is an interface between platform owner and third-party developers (Evans et al., 2008; Hanseth & Lyytinen, 2010; Messerschmitt & Szyperski, 2005). According to Ghazawneh and Henfridsson, 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" (Ghazawneh & Henfridsson, 2013 , p.174). SDKs, APIs, guidelines, documentation, incentives and IP are typical examples of platform boundary resources. Platform owners shift selected design capabilities to developers by designing the boundary resources properly. Star and Griesemer (1989) articulated that boundary resources should be "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 & Griesemer, 1989, p. 393). In addition, boundary resources facilitate access to core functionalities of the platform to serve end users by developing a wide range of innovative applications (Bergman, Lyytinen, & Mark, 2007; Yoo et al., 2010).

Given the importance of network effects, when it comes to platform success and survival, the external developers play significant roles (Taudes, Feurstein, & Mild, 2000). Independent developers contribute to the platform's ecosystem by entering into agreements proposed by the platform. These agreements are also viewed as boundary resources.

Acknowledging the supporting role of third-party developers in expanding the platform ecosystem, the firm tends to offer developers useful and generative tools and resources. One approach to doing so can be initiated by studying the internal resources, such as APIs. In this phase, the firm should have an acceptable understanding of the resources' value and the potential value that these resources can generate for the ecosystem. In other words, it is crucial for the firm to understand different aspects of internal resources' transformation or process of exposing into the boundary resources. When analysing the value of the internal resources and potential value of the same resources when they are available to the third-party developers as platform boundary resources, positive aspects and negative consequences are some of the critical dimensions that should be clearly discussed for the firm.

3 Value Modelling

As previously mentioned, to understand the value of a platform owner's asset, a detailed analysis of the enterprise is required. A well-defined method should be used to define, derive and analyse different business scenarios and relationships within the enterprise.

3.1 Value Modelling Tools in Practice

A number of value modelling methodologies have been used in practice to visualize the value of an asset inside an organization. Literature discuss different methodologies and modelling techniques for software and service development such as Goal Modelling, Workflow modelling and value modelling. Recent studies highlight the implication of these methodologies in Strategic API assessment.

3.2 E3 Value Modelling Tool

When it comes to value modelling methodologies, literature suggest some practical tools that can be used in practice to model economic value in business ecosystem setting. One of these methodologies is E3 Value Modelling tool[2] (Gordijn & Akkermans, 2003). This value modelling tool has been used for different purposes such as economic value analysis or modelling value flow in a network. However, this tool has not been used to visualize the potential value of APIs. In this study, we use a conceptual model called e3 value modelling (Gordijn & Akkermans, 2003) to illustrate and visualize how value is shaped and exchanged among the actors in a network. This methodology affords the firm to model visual and conceptual understanding on the value of the assets in the ecosystem.

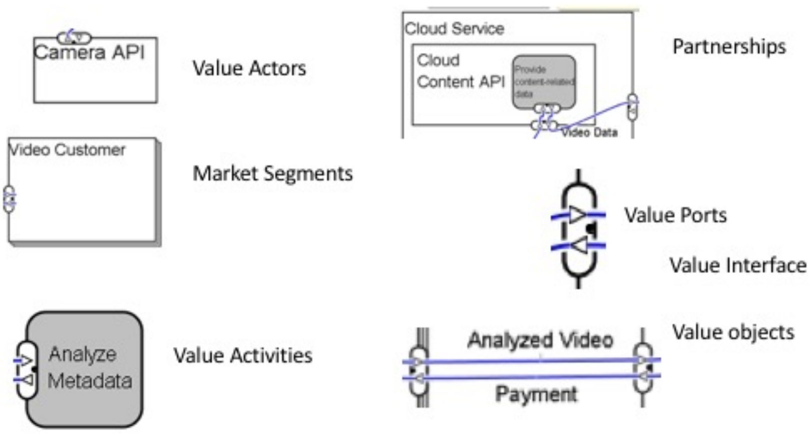

Figure 1: E3 value mapping syntax with examples[3] (Horkoff, 2018)

[2] http://e3value.few.vu.nl

[3] This is an illustrative example of E3 value modelling notation from our earlier works in the same project.

Actors in the network are defined as independent economic entities that should potentially be capable of making a profit via value activities. Services, products or money are different types of value objects that actors exchange. Value ports can be used by the actors in case of request or provide value objects, and finally, *"A value interface shows the value object an actor is willing to exchange in return for another value object through its port*s" (Gordijn, Akkermans, & Van Vliet, 2001 , p.11).

We will not adhere or subscribe to the full formal definitions of this method and instead extend and modify the original method in several ways by making it "lighter." In what follows, we argue why such a lightweight version of this methodology is useful for discussing several of the questions in focus here.

Developed originally in the 2000s, e3 value modelling is based on the combination of business value and the formal systems theory (Gordijn, Akkermans, & Van Vliet, 2000). e3 value modelling was originally used to discuss e-business scenarios and value flows that were relatively easy to quantify, focusing primarily on economic value. Later, there were research efforts to extend the method to discuss, for example, ecosystem actors (Leimeister, Böhm, Riedl, & Krcmar, 2010) and e3 (Razo-Zapata, Gordijn, De Leenheer, & Wieringa, 2015).
The original e3 value methodology has several main characteristics that make it very suitable for discussing APIs. First of all, it focuses on analysing economic value, i.e., creation, exchange and consumption of economically valuable objects such as services, products, currency and experiences in a multi-actor network. Secondly, e3 is built to discuss requirements engineering from different perspectives and provides semi-formal conceptual models. Thirdly, the concept of a value port is a way for an actor to indicate the willingness to provide or request value objects: It enables reducing focus on the business process and instead focusing on how (external) actors can connect to the network (for a more thorough discussion of the method in an embedded ecosystem (Leimeister et al., 2010).

Gordjin lists the steps of the model in the e3 summary handbook (Gordijn, 2002), discussing the following steps to visualize the value: 1) have an idea for a service, 2) construct a value model for this idea (actors, the objects of value created, distributed and consumed), 3) find variations (deconstruct into smaller pieces and rebundle), 4) engage in multi-viewpoint exploration and 5) evaluate profitability. Each step can be broken down into further activities; for example, step 2 can be broken down into further steps, which include taking the existing customer needs perspective as a starting point, identifying relevant actors, focusing on value exchanged, identifying value ports and discussing choices and incentives for the actors.

We want to stretch the value modelling to discuss and visualize APIs and, more specifically, value-related questions about APIs. This API focus means that we have made a number of modifications to the restrictions provided by the language: We take an inclusive approach to value, letting the organization define values and also allowing "softer" values to be modelled. We also discuss internal exchanges of value (i.e., internal APIs), and we allow for one-directional flows. According to our investigation, E3 value modelling and a suitable method to develop valuable API to the ecosystem were missing in the organization's toolbox.

4 Methodology

To answer the research question, we have conducted an exploratory in-depth case study on a project called API Strategies with an ambition to "provide industrial partners with support for how to build an

API strategy that involve both internal actors and external stakeholders."[4] Within the knowledge-building process, a case study is a suitable method for the exploration, classification and hypothesis formation phases (Benbasat, Goldstein, & Mead, 1987). As described by Yin (2013), single case studies suit research with a representative case, where the researchers can capture the different aspects of typical situations (Yin, 2013). In addition, we choose single case study approach as one of the main rationales for choosing single case is when the researchers access to a representative case that can reflects different circumstances of typical situation (Yin 2009). In fact, from our perspective this case that we study is illustrative enough to describe the results. The research was carried out in an academy-industry collaboration network that spans several universities.

4.1 Research Setting

The work in the collaboration network is organized into dedicated teams focusing on specific concerns of the industrial partners. The project was designed to investigate in detail the API strategies in which the different organizations were engaged related to their internal API value chain. Work in the network is divided into 6-month sprints, and the current presentation of findings sums up the work carried out over 21 months. The focus of the analyses was on the aspects related to API strategies because they use a variety of APIs. The research project involved several organizations and their API strategies, but in this paper, we focus on only one of the research partners because it was deemed especially interesting for the purposes of their API management. The organization provided us with very good access to specific APIs for the purposes of conducting this study. The company's customer base consists of both private- and public-sector organizations. Currently, several APIs are provided to their customers, partners and industrial developers. For example, cloud APIs are provided to increase customer loyalty. The international compliance environment is challenging. The organization was involved with both internal, consortium-based and external (open) APIs.

4.2 Data Collection

Each one of the sprints started with a coordinating workshop for the project partners. This was followed by a series of group and individual workshops and interviews at the company premises to collect the qualitative data needed in the modelling. Unclear things were followed-up via email. Each meeting was recorded for future use.

Company workshops normally had around 4-5 participants. Participants were normally from technical roles and had some familiarity with software modelling.

Data collected from workshops and interviews (including documentation) was used to generate the models. Workshops lasted three hours and individual interviews were normally around one hour. Part of the modelling was carried out as a thesis project where the research and models were discussed internally in a weekly research meeting.

4.3 Data Analysis

In the workshops, selected APIs were first discussed among the project members. Then, the project team, used several different kinds of discussion and visualization techniques to map out the strategy-

---

[4] https://www.software-center.se/research-themes/technology-themes/customer-data-and-ecosystem-driven-development/api-strategies/

related activities of the company regarding the specific API and API environment. Example tools discussed and tried out in the workshops included modelling of workflow, goal model and different value models.

As previously mentioned two authors in this study participated in the project workshops and meeting. According to our observations, the way of working was reflective and iterative: Work was carried out in consecutive rounds. Feedback on the method and process was gathered at the same time as a group of researchers in the project team provided the visualizations for discussion among the participants. The information on the APIs of interest were collected over several workshops and provided the early drafts of the value models for discussion. A number of different discussions and assignments were also carried out to discuss API value in these workshops.

In what follows, we discuss the findings from these workshops regarding value modelling in more details.

5 Results

We posed the following research question: "How can platform owners identify and visualize values of potential boundary resources?" We have noticed that the project team was interested in value analysis techniques. According to existing knowledge, competences and experiences among the project team members, they wanted to investigate whether e3 value modelling would be a suitable technique, as explained, to better understand the strategic value of (internal and external) APIs in a complicated ecosystem.

As a result, the project team provide a methodology for API strategy identification, visualization and reflection.

Stage	End Result	Activities
1. Identification	**Selection**	Select stakeholders to participate
1. Identification	**Selection**	Select an API
1. Identification	**Selection**	List values related to the API
2. Visualization	**Value models**	Draw the entities
2. Visualization	**Value models**	Draw partnerships
2. Visualization	**Value models**	Draw value ports and flows
3. Reflection	**Priorities**	Collect the value flows

Table 1; Table 1 Process of API value modelling

5.1 Results Identification

In this chapter each stage of API value modelling will be defined in details. The first stage of the process of value modelling is the identification of relevant APIs, stakeholders and values to be modelled. The first activity of the identification stage is to **identify the API**(s) on which to focus. This selection can be done for a newly designed API or an existing API. In the workshops, the project team discussed new APIs and existing APIs in the beginning of their life cycle. The Organization was in charge of selecting the APIs.

The second activity is to **identify stakeholders** that should be involved in the process of visualization. We noticed that this is a balancing act – more stakeholders will likely give more insight from different perspectives on one hand, but on the other hand, bringing many people in will increase the cost of the modelling effort and make it harder for the participants to participate actively in the modelling effort.

The third activity is to list **values related to a** specific **API.** There were two different ways this could be accomplished. We have realized that the project team could have 1) presented a large number of identified values, metrics or goals from the earlier literature to start the discussion, or 2) started from a specific API and worked their way up to discussing the most important aspects related to values. The project team in this study, selected the later method in the workshop and identified values such as support and training, equipment sales, maintenance and upgrades, services, enriched content, trust, collected customer data and so on. When looking at the collected values, the team noted that "value" was interpreted very broadly in the discussion. Almost anything of use was listed as a value. The team wanted to be inclusive at this point and did not go in depth into questioning the nature of the values elicited or whether they actually were values. They also noted that several values were very high-level and strategic, which would make them difficult to measure.

The results of this stage are the selection of the API to model, a list of relevant stakeholders that need to be included and a list of values related to the API.

5.2 Results Visualization

The second stage is **visualization**, which can be further broken down into drawing the entities, partnerships, value ports and value flows. This visualization can be done in different kinds of settings: virtually or non-virtually, in a group or individually, using specific modelling tools or drawing on a blackboard. To engage the stakeholders, in this study, the project members used a non-virtual group setting and worked by drawing on some prepared material.

The first activity of the visualization is to **draw the entities**. The team started this exercise from the perspective of the API and listed the most relevant stakeholders as organizations, business units and user groups who are directly working with the API.

The second activity is **drawing the partnerships**. In this case, partnerships were used to indicate boundaries of organizations. The different business units were grouped inside the same overall company to indicate that the API was internal.

Then, the project team **drew the value ports and flows** from one entity to another. This was based on work carried out in the first stage, where they had identified the values. When doing this activity, we noted missing entities or novel ways to break the existing entities. According to the original methodology, the team was expecting to see some reciprocity of value exchange, but for many internal APIs, they only identified one-directional flows.

The result of the second stage is the visualization of the API, including entities, partnerships and ports and named value flows between the entities.

5.3 Results Reflection

Stage two results in value models that are the starting point for the third stage: reflection. This was divided into two separate activities: collecting the value flows and discussing how to prioritize these value flows.

Collecting the value flows included going through the generated models and discussion and listing the different values that emerged from those models and discussions. The values were then grouped together and presented to the organization.

The final activity is to **discuss how to prioritize** the different values, drawing on the models. Different criteria for prioritization might be applied here: For example, emphasis could be placed on immediate economic value, growing the ecosystem or identifying the minimum requirements needed.

The result of this third stage is an improved selection of values related to specific APIs and their priority among the selected values. This gives the organization a shared visualization, conceptualization and understanding of the API and related value questions.

The project team also collected feedback at the end of the session to improve the method and to better understand any concerns the participants might have had.

5.4 Simplified example of value model

Figure 2. illustrates an example of the team's analysis using E3 Value modelling tool. In this figure we can see how the tool visualized the actors, value object, value activity and value flow within an ecosystem. This figure is a illustrative example that has been simplified for this purpose.

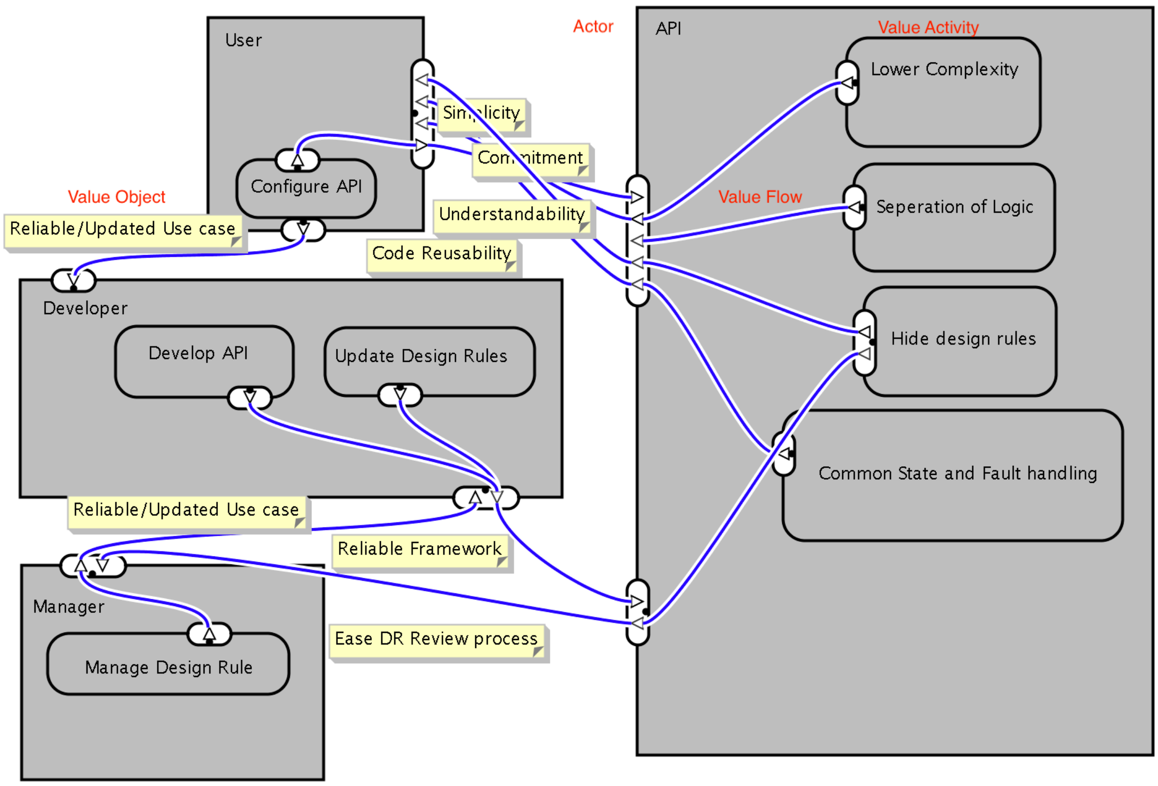

Figure 2: Simplified Example of Value Model

Our example API enforced software design rules but was under used. Value modelling was used to visualize the value of the API in this setting.

6 Discussion

This research study shed some light on how platform owners can identify and visualize values of potential boundary resources. While understanding the value of resources from developers (who are truly users of the boundary resources) is significant, the focus of this research is on platform owners' challenges in providing boundary resources. Many studies that promote openness as a response to organizational issues in competitive markets highlight the benefits of introducing internal assets to the external actors in the ecosystem. In an effort to contribute to this area of research, we studied a company's approach to assessing their resources' value. This approach can be described in 3 phases: identification, visualization and reflection. In this study, e3 business value modelling has been suggested as an assessment tool to assist the firm in understanding the value of the assets that they already have for the firm, current users of the assets (internal developers, external developers or contracted partners with specific agreements), the value that it can create for the ecosystem if exposed as a boundary resource to third-party developers and the potential value, negative effects and related consequences to the firm's current business model. The first step, identification, helps the platform owner to think about and filter out the most important partners, existing assets and their values in a more structured way. Then, the value modelling tool facilitates understanding the direction of value flow between partners in the ecosystem. Without a clear picture of the values for the ecosystem, platform owners would have a hard time fulfilling the demands. Finally, the clear value model would assist the decision-makers in the firm regarding whether to keep an asset internally or provide it as a boundary resource to the ecosystem.

Understanding the current value of internal APIs and the potential value of these assets in the case of transformation into boundary resources plays a significant role in platform owners' decisions regarding whether to keep the internal API as a competitive advantage in the market or offer the third-party developers a generative tool and boost innovation in the ecosystem.

We also realized that prioritizing the assessment criteria in this study is meaningful. These criteria may differ from organization to organization as the business models are different. Prioritizing some criteria over others would facilitate mapping the value flow.

In addition, this research shows that the concept of boundary resources can be defined according to the firm's definition of the boundaries. In other words, the e3 business value modelling suggests that some assets should be offered as boundary resources, while other resources are better to be kept as internal APIs and other resources should be exposed to specific groups of partners in the ecosystem. This means that while specific partners in the ecosystem may access some APIs, these resources are not available for others. This different level of access to the resources can also be defined by the suggested framework.

7 Conclusion

By using the suggested e3 business value mode, platform owners can identify and visualize the value of their resources that can potentially be transformed into platform boundary resources. In this investigation, we have identified a number of limitations. First, we should admit that in this research, the role of developers was still rather passive, even though business unit managers and application developers took part in our workshop. In other words, in this study, we focused more on platform owners' actions and decisions. To have a broader view on the value of boundary resources, developers' opinions should also be put on the table. In addition, in further research, E3 value modelling can be evaluated in a multiple case setting.

The second point is also related to the developers' opinions on the boundary resources. Because platform owners and developers may have very different perceptions of values and services, the value of the boundary resources may differ between these two different sides' viewpoints. Consequently, tuning of platform boundary resources might be needed after presenting them to the ecosystem and engaging developers in practice and use of the resources.

Finally, it is also notable that while e3 business value modelling is suitable and transferable across contexts to assess the value of the potential boundary resources, the exact evaluation criteria can differ from case to case, platform to platform and ecosystem to ecosystem.

Acknowledgements

Thanks to company contacts and the Chalmers Software Centre for support.

References

Barney, J. (1991). Firm resources and sustained competitive advantage. *Journal of management, 17*(1), 99-120.

Benbasat, I., Goldstein, D. K., & Mead, M. (1987). The case research strategy in studies of information systems. *MIS quarterly*, 369-386.

Benkler, Y. (2006). *The wealth of networks: How social production transforms markets and freedom*: Yale University Press.

Bergman, M., Lyytinen, K., & Mark, G. (2007). Boundary objects in design: An ecological view of design artifacts. *Journal of the Association for Information Systems, 8*(11), 546.

Bolici, F., Howison, J., & Crowston, K. (2016). Stigmergic coordination in FLOSS development teams: Integrating explicit and implicit mechanisms. *Cognitive Systems Research, 38*, 14-22.

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), 1409-1427.

Brynjolfsson, E., & Saunders, A. (2009). *Wired for innovation: how information technology is reshaping the economy*: MIT Press.

de Souza, C. R., & Redmiles, D. F. (2009). On the roles of APIs in the coordination of collaborative software development. *Computer Supported Cooperative Work (CSCW), 18*(5-6), 445.

Eaton, B., Elaluf-Calderwood, S., Sorensen, C., & Yoo, Y. (2015). Distributed tuning of boundary resources: the case of Apple's iOS service system. *MIS quarterly, 39*(1), 217-243.

Evans, D. S., Hagiu, A., & Schmalensee, R. (2008). *Invisible engines: how software platforms drive innovation and transform industries*: MIT Press.

Ghazawneh, A., & Henfridsson, O. (2013). Balancing platform control and external contribution in third-party development: the boundary resources model. *Information Systems Journal, 23*(2), 173-192.

Gordijn, J. (2002). *E 3-valuein a Nutshell.* Paper presented at the Proceedings of International Workshop on E-business Modeling, Lausanne.

Gordijn, J., Akkermans, H., & Van Vliet, H. (2000). *What's in an electronic business model?* Paper presented at the International Conference on Knowledge Engineering and Knowledge Management.

Gordijn, J., Akkermans, H., & Van Vliet, J. (2001). Designing and evaluating e-business models. *IEEE intelligent Systems, 16*(4), 11-17.

Gordijn, J., & Akkermans, J. (2003). Value-based requirements engineering: exploring innovative e-commerce ideas. *Requirements engineering, 8*(2), 114-134.

Hanseth, O., & Lyytinen, K. (2010). Design theory for dynamic complexity in information infrastructures: the case of building internet. *Journal of Information Technology, 25*(1), 1-19.

Horkoff, J., Hammouda, I., Lindman, J., Knauss, E., Debbiche, J., Freiholtz, M., Liao, P., Mensah, S., Stromberg, A. (2018). Modeling support for strategic api planning and analysis. *n the proceedings of ICSOB 2018*.

Leimeister, S., Böhm, M., Riedl, C., & Krcmar, H. (2010). *The Business Perspective of Cloud Computing: Actors, Roles and Value Networks.* Paper presented at the ECIS.

Messerschmitt, D. G., & Szyperski, C. (2005). Software ecosystem: understanding an indispensable technology and industry. *MIT Press Books, 1*.

Mohagheghzadeh, A., & Rudmark, D. (2017). *Accelerated Tuning of Platform Boundary Resources.* Paper presented at the Scandinavian Conference on Information Systems.

Mohagheghzadeh, A., & Svahn, F. (2016a). Shifting Design Capability to Third-Party Developers: An affordance Perspective on Platform Boundary Resources.

Mohagheghzadeh, A., & Svahn, F. (2016b). *Transforming Organizational Resource into Platform boundary Resources.* Paper presented at the ECIS.

Penrose, E. T. (2009). *The Theory of the Growth of the Firm*: Oxford university press.

Prügl, R., & Schreier, M. (2006). Learning from leading-edge customers at The Sims: opening up the innovation process using toolkits. *R&D Management, 36*(3), 237-250.

Razo-Zapata, I. S., Gordijn, J., De Leenheer, P., & Wieringa, R. (2015). e 3 service: A Critical Reflection and Future Research. *Business & information systems engineering, 57*(1), 51-59.

Shapiro, C., & Varian, H. R. (1999). The art of standards wars. *California Management Review, 41*(2), 8-32.

Star, S. L., & 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), 387-420.

Svahn, F., Lindgren, R., & Mathiassen, L. (2015). *Applying options thinking to shape generativity in digital innovation: An action research into connected cars.* Paper presented at the System Sciences (HICSS), 2015 48th Hawaii International Conference on.

Taudes, A., Feurstein, M., & Mild, A. (2000). Options analysis of software platform decisions: a case study. *MIS quarterly*, 227-243.

Tiwana, A., Konsynski, B., & Bush, A. A. (2010). Research commentary-Platform evolution: Coevolution of platform architecture, governance, and environmental dynamics. *Information Systems Research, 21*(4), 675-687.

Wernerfelt, B. (1984). A resource-based view of the firm. *Strategic management journal, 5*(2), 171-180.

Yin, R. K. (2013). *Case study research: Design and methods*: Sage publications.

Yoo, Y., Henfridsson, O., & Lyytinen, K. (2010). Research commentary-The new organizing logic of digital innovation: An agenda for information systems research. *Information Systems Research, 21*(4), 724-735.