

ESAO: A Holistic Ecosystem-Driven Analysis Model

Jan Bosch¹ and Petra Bosch-Sijtsema²

¹ Chalmers University of Technology, Department of Computer Science and Engineering,
Software Engineering, Gothenburg, Sweden

² Chalmers University of Technology, Department of Civil and Environmental Engineering,
Construction Management, Gothenburg, Sweden
{Jan@JanBosch.com, Petra.Bosch}@chalmers.se

Abstract. The growing importance of software ecosystems and open innovation requires that companies become more intentional about aligning their internal strategy, architecture and organizing efforts with the ecosystem that the company is part of. Few models exist that facilitate analysis and improvement of this alignment. In this paper, we present the ESAO model and describe its six main components. Organizations and researchers can use the model to analyze the alignment between the different parts of their business, technologies and ways of working, internally and in the ecosystem. The model is illustrated and validated through the use of three case studies.

Keywords: Ecosystem, Strategy, Architecture, Organizing, Model.

1 Introduction

Recently, more and more research discusses the relevance of ecosystems for companies as well as market segments. This research lifts up the business aspects of ecosystems, the innovation element as well as the more technical component of ecosystems. An ecosystem is defined as an economic community supported by a foundation of interacting organizations and individuals, which can also be perceived as the organisms of the business world [1]. The terminology of business ecosystem defines ecosystems as consisting of three characteristics [1, 2, 3]: (a) a symbiosis relationship in which the survival of all members implies the survival of the ecosystem. (b) Co-evolution in which partners co-evolve capabilities around new innovations and finally (c) ecosystems are often based on a particular platform, which is defined as tools, services or technologies used in the ecosystem that enhance performance of its members [4, 5, 6]. Especially in the software industry the term software ecosystem has gained enormous popularity and can be defined as: a software ecosystem consists of a software platform, a set of internal and external developers and a community of domain experts in service to a community of users that compose relevant solution elements to satisfy their needs [7, 8].

Current ecosystem research primarily looks at the ecosystem – but does not link this back towards the internal organization or to the implications of the internal organization, software platform or architecture and the ways of organizing and

working. This is a challenge for many organizations for several reasons. First, the way the organization works internally and the way the company engages with its ecosystem need to be closely aligned with each other, as a strong co-dependency exists between the two. Second, as companies increasingly seek to focus their internal efforts on what truly differentiates them and try to outsource as much as possible of the non-differentiating activities, the reliance on the ecosystem is increasing significantly. Finally, as customers are transitioning from buying products to acquiring services, the burden of integrating different products into a dedicated solution for customers is falling to ecosystem players that need intimate interactions with other players in the ecosystem. However, few, if any, holistic models exist that support organizations to gain a better insight for aligning the complex internal and ecosystem dimensions of R&D.

In this article we propose a model that encompasses elements of both the ecosystem as well as the internal organization. The model provides three perspectives, i.e. strategy, architecture and organizing, and two dimensions, i.e. internal and ecosystem. This provides an approach where the alignment between the perspectives and between the internal and ecosystem dimensions can be analyzed and improved.

The remainder of this paper is organized as follows. Below we first describe the problem statement. In section 3, we introduce the ESAO model and describe its six main components. Section 4 is concerned with validating the ESAO model with three case study examples where we illustrate the alignment, or lack thereof, between the six components of the ESAO model. Finally, we conclude the paper and discuss the core contribution and its implications.

2 Problem Statement

With the growing proliferation of open innovation and software ecosystems, it becomes more and more important that software research takes a more holistic picture instead of describing individual elements. The vast majority of research studies one dimension of a software intensive organization. These different dimensions are, for example, the software ecosystem, the architecture, as well as the organizational aspects of software intensive organizations. The software ecosystem literature primarily studies the ecosystem and discusses different roles in the ecosystem as well as a strong focus on the keystone player in a particular ecosystem [8]. In the software architecture literature there is a strong focus on the design and evolution of the architecture as well as on the technology choices required to support software architecture [9]. In the agile community there is a lot of focus on how to organize agile teams as well as processes and ways of working [10, 11, 12]. However, virtually all research takes a narrow and deep approach, focusing on a specific aspect. In our review of the literature, we found very few models that provide a holistic and complete overview of all these different dimensions and their interdependency.

One often applied model that provides a holistic perspective of the end-to-end dimensions of business, technology and organization is the BAPO model [9, 13]. The BAPO model defines for four independent software development concerns: (1)

Business, concerned with how to make a profit, (2) Architecture, concerned with the structure of and technologies required to build and evolve the software system, (3) Process, defining the roles, responsibilities and relationships within software development as well as the tooling and ways of working and, finally, (4) Organization, defining the actual mapping of roles and responsibilities to organizational structures [13]. The model is frequently applied for analysis and assessment in both academia and industry. In the figure below, the BAPO model, as well as the intended dependencies in the model, are shown.

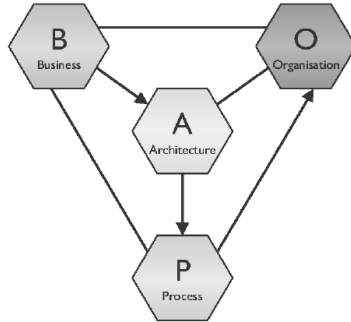


Fig. 1. The BAPO concerns (source: [13]).

One of the authors of this paper was involved in the development of the original BAPO model. However, in ten years of evolving an understanding of the problem, we can identify a number of challenges with the original BAPO model. These challenges are the following:

- BAPO is a model that only incorporates the internal organization, but does not take into account the external environment like the ecosystem. From software ecosystem literature we know that the ecosystem has a major impact on an organization [7] and therefore it is important that an ecosystem dimension is part of such a model.
- The BAPO model was originally developed in the context of software product line research and in its detailed definition assumes domain software and product software. This limits the applicability of the model in practice as not all companies are using software product lines. Although not impossible, the model is less applicable to companies that are less reuse-centric.
- BAPO strongly enforces a $B \rightarrow A \rightarrow P \rightarrow O$ sequence. In practice, however, this represents too much of a simplification of the reality in the organizations that we engage with. Even though idealistically speaking the sequence of the B (business) should drive the A (architecture), A should drive the P (processes) and P should drive the O (organization structure), in practice one never starts from a green field situation. Consequently, one has to allow for bi-directional dependencies and the focus should be on achieving alignment between the four dimensions.

Having analyzed these challenges, we came to the conclusion that there is a need for a new improved model that allows organizations to accomplish the following:

- Serve as a holistic analysis framework
- Serve as a benchmark for effective software product engineering
- Support the assessments of software product engineering for capability
- Evaluate software production units, divisions, or companies
- Support the improvement of software product engineering, which involves producing assessments and improvements plans

The list above is a quite extensive list of requirements on any model. In this paper, however, we focus specifically on the analysis model, i.e. the first item, and its facilitation of alignment between the different aspects of a software-intensive systems organization.

3 The ESAO Model

Based on the discussion above, we propose an extension and evolution of the BAPO model, i.e., the ESAO model. The model is based on our experience from working with dozens of companies around R&D management topics. As we have outlined in the earlier parts of the paper, we have increasingly identified challenges with the tools that we had available to help companies and in response, we have developed a new model. The ESAO (Ecosystem, Strategy, Architecture and Organizing) model consists of six interdependent and interconnected dimensions that are important to take into account for software development. The six dimensions of the ‘ESAO’ model concern both an internal company and an external company perspective.

In the remainder of this section, we first describe the internal perspective and subsequently we discuss the ecosystem perspective.

3.1 Internal Perspective (SAO)

The internal perspective consists of three main dimensions, i.e. strategy, architecture and organizing. Below, each of these dimensions is defined in more detail.

1. ***Internal Company Strategy***: The strategy of the company lays down the basis for the future path of the firm concerning the business. In particular, the strategy is concerned with how the company generates revenue now and in the future. The company strategy is relevant for the internal prioritizations and decisions made within an organization, and is closely related to the software development strategy and architecture. The internal business model development is part of the internal strategy. The business model defines how the firm creates and delivers value to customers and then converts payments received to profits [14]. The internal company strategy can be related to the Business concern of the BAPO model.

2. **Internal Architecture:** The architecture comprises the technical structure means to build the software-intensive system as well as the technology choices. The company strategy defines which aspects of the business will develop going forward and need to be prioritized and which can be deprioritized. This is important input for the architecture decisions as it allows effective management of future evolution cost. The internal architecture dimension can be related to the Architecture dimension of the BAPO model.
3. **Internal Organizing:** The ways of organizing work, way of working, roles, responsibilities, processes and tools within software development are important and closely related to the architecture and strategy of the firm. In an earlier publication one of the co-authors developed the concept of ‘Stairway to Heaven’, to describe how development typically evolves over time [16]. This element is related to the Process and partly Organization part of the BAPO model. The ESAO model combines the P and O parts of the BAPO model as, in practice, the adoption of agile approaches assumes empowered, cross-functional teams and the locus of power is much more with the teams than with the traditional reporting hierarchies. As a consequence, the precise organization structure is less important than earlier and the focus has shifted to organizing the work.

3.2 External Ecosystem Perspective (ESAO)

In the ESAO model, we use the same three dimensions discussed above for the external ecosystem. However, depending on the role of the organization in the ecosystem, the company has more or less power in its ecosystem. When discussing the ecosystem as an important dimension in relation to the internal strategy, software architecture and way of organizing, it becomes relevant to understand that firms can obtain different roles within an ecosystem. These roles are often discussed and defined in literature and are also lifted up in the next section concerning validation of the model with help of three case studies. The main roles often studied in ecosystems are the following:

- Central firm or also called the keystone or platform firm who is the dominant player and orchestrator in the ecosystem [2, 3, 4, 5, 6, 8].
- Complementors and component players who provide a product or service that complements the platform or product of an ecosystem and enhances the value of the platform [5, 8].
- Integrators who brings together the parts provided by different ecosystem players into an integrated solution for the end-user. Depending on the ecosystem, this role can be played by the keystone player, the end-user or a separate organization [5, 8].
- A final role important in the software ecosystem is the end-user [8].

The role that the organization plays in its ecosystem determines the amount of freedom that it has in terms of defining its strategy, architecture and organizing

dimensions. However, even complementors should not view themselves as powerless. Instead, every player has a set of strategic options available to optimize their position and future outcomes:

1. ***Ecosystem Strategy***: The external strategy of a company is related to the business and software ecosystem of the firm and the strategic options that it has available in its current role in the ecosystem. As a keystone player strategic decisions are concerned with providing a viable business model for complementors while maximizing its own revenue. In addition, whether the complementors should be encouraged to compete or if the focus should be on collaboration (cf. [15]). For complementors, the goal often is to maximize its own stake in the ecosystem. One strategy is to seek to form a niche market in the ecosystem, to become the keystone partner in that niche and to expand from that position of strength. For integrators, the relationship to the end-user and maximizing its own visibility while diminishing the role of other ecosystem players is often a viable strategy to increase its relevance. Depending on the strategic choices made by the company, there are significant implications on the system and software development of the firm.
2. ***Ecosystem Architecture***: The ecosystem architecture defines the interface between the internal architecture and the solutions that are provided by ecosystem partners in terms of the following:
 - a. The interface between my firms suppliers and my firm
 - b. The interface between firms that build software on top of my product or platform. These roles are also discussed as complementor roles [5, 8] and they can deliver, add, or develop components and complements to a product or platform as a complement to your firm's platform or product.
 - c. The interface between my firm and firms that operate in the same ecosystem role as my firm, but that provide other types of functionality.
 - d. Finally, depending on the player providing the integration of ecosystem solutions, the interface between my firms and integrators.

In addition to the focus on interfaces, the focus is also on the architecture strategy. As we discussed in [17], there is a constant commoditization process ongoing that requires that ecosystem players pro-actively innovate around new functionality and release commoditizing functionality to other players or the open-source community.

3. ***Ecosystem Organizing***: Deals with how firms work with their customers, suppliers, and ecosystem partners in terms of processes, tools used, ways of working, and ways of organizing the collaboration. For instance, in some of the companies that we work with, the company has internally adopted agile ways for working and continuous integration. However, the suppliers of the company still use traditional waterfall or iterative development causing the supplied parts of the system developed by the company to be updated very

infrequently. This causes significant disruptions in the internal development processes.

In the figure below, we present the ESAO model graphically (figure 1). We aim to illustrate two important aspects. First, in order to optimize synergy, alignments between the internal and the ecosystem dimension, the strategy, architecture and organization perspectives become important. For instance, a business strategy that stresses high responsiveness to customer requests does not combine well with a traditional, iterative development approach with new releases of software every six months.

Second, equally important, the internal and external perspective on each dimension should be in line with each other, i.e. ecosystem strategy and internal strategy, ecosystem architecture and internal architecture as well as ecosystem organizing and internal organizing. For instance, the aforementioned example of the company using agile methods internally and waterfall methods with its suppliers illustrates the inefficiencies caused by lack of alignment.

A change in one of the dimensions has consequences for the other dimensions. Although no organization will every be completely aligned in all six dimensions of the ESAO model, our experience has shown that there is significant benefit to continuously seek alignment when implementing changes in one of the six dimensions. That allows misalignment to be short-lived and affecting the organization as little as possible.

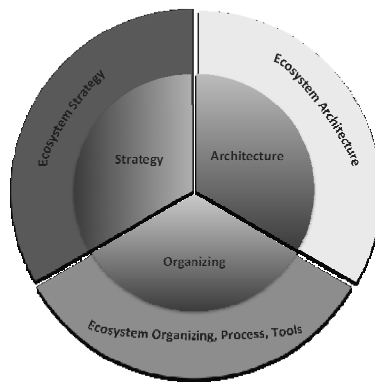


Fig. 2. The ESAO model with internal and ecosystem dimensions.

The ESAO model does not insist on a particular order, but instead focuses on achieving and maintaining alignment between the different dimensions. The model could be related to the research methodology of systems thinking which encompasses a holistic approach in which often an external and internal focus is taken into account. However, the system that is studied can vary depending on the definition and boundaries set for a system. The ESAO model provides a framework that supports the analysis of both the external and internal firm.

4 Validation

We validate the use of the ESAO model with the help of three individual cases. In all three cases either the ecosystem changed or the position of the firms in their ecosystem changed and this had implications for the internal as well as ecosystem strategy, architecture as well as way of organizing in the different firms. With help of the cases we show that being able to analyze, assess, and react to the external dimension, i.e., the ecosystem dimension is important for firm's strategy, internal development and way of working. Below we use the ESAO model as an analysis framework for three different cases in which we introduce the case, discuss the change trigger and analyze the case according to the ESAO dimensions. Due to limitations in space of this paper, we can only describe the findings and analysis rather briefly. The examples described below are extracted from longer-term data collection through interviews, workshops and discussions between researchers and managers and R&D engineers. For case study Alpha we held 13 interviews in the firm, and had numerous workshops with complementors and end-users (total of 21 people). In case Beta we primarily held 4 group interviews and workshops with 5-10 people (total of 20 people) and in case Zeta we held 14 group interviews with 5-10 people (total of 50 people). Three case studies cannot give sufficient generalization for the model, but give an insight in how the model can be applied to analyze the complexity of software development R&D. In the anonymous cases we discuss below we show that a change in one of the six factors has implications for the other ESAO factors. The interviews were held retrospectively in order to capture the full implications of these external and internal changes. The companies were therefore selected based upon the fact that strategic changes were implemented in their firm.

4.1 Case Alpha: Ecosystem Changes

Introduction to Case Alpha. Company Alpha is a Fortune 1000 company developing software products and services operating, primarily, on personal computers. The company's products address both consumer and business markets and the company releases several products per year, including new releases of existing products and completely new products. The products developed by the company range in the multi- to tens of millions lines of code and tend to contain very complex components that implement national and international regulations. The case concerns one of the products of the company that has a user base of millions. For this product, the company is the keystone or dominant player in their ecosystem.

Change Trigger. The case study is dealing internally with a changing ecosystem. The company had treated its entire customer base as a relatively homogeneous population; however, based on market research and customer feedback it became increasingly clear that many customer segments existed with unique and specific needs. On the other hand with a customer base numbering in millions, there was a growing base of developers of both within case Alpha as well as outside of the firm (i.e. complementors), that in various non-endorsed ways sought to extend the functionality

in the base product with features for individual customers or narrow customer segments. From the interviews it was stated that the company could never serve all these segments in a cost effective manner.

ESAO Analysis

Ecosystem Strategy: the case company has a keystone role within their ecosystem concerning this particular product. For many years the company either ignored or actively discouraged third party developers to extend its product. A number of years ago with the advent of the iPhone apps the company decided to adopt an alternative ecosystem strategy. It decided to copy the Apple app store model and collect 30% of the sales generated by 3rd party developers.

Ecosystem Architecture: the change in ecosystem strategy caused the architects to introduce an ecosystem API to the product. However, as the product managed quite sensitive data for its users, the company introduced a multi-layered API where certified apps would get more access, and non-certified apps only received read-only access.

Ecosystem Organizing: the company decided to pro-actively engage with its developer community through the organization of developer conferences, regular newsletters and other forms of communication. In addition, the company created a certification mechanism that allowed 3rd party developers to certify their application. Finally, the company introduced a market place inside its product that managed payments (inside apps) as well as entitlement for 3rd party developers.

Internal Strategy: the predominant change in business strategy for the company concerned the best ways of serving customer segments. Before the adoption of the ecosystem strategy, the ongoing debate within product management, concerned the introduction of customer segment specific functionality, versus the increased complexity of the product for customers in different segments. After the adoption of the ecosystem strategy, interviewees mentioned that the discussion changed and focused on the boundary between functionality that should be in generic products within the platform and functionality that should be left to the developer community.

Internal Architecture: the impact on the product architecture is two-fold.

- (1) The ecosystem API was introduced which required a careful analysis of which parts of the product internals were to be exposed and which would remain hidden.
- (2) As the company adopted a certification mechanism, the ecosystem API, as well as the rest of the product, had to support the differentiation between certified and non-certified apps.

Internal Organizing: the primary change in the internal organization was the development of a unit responsible for 3rd party developers. This unit was both responsible for certification of apps, as well as for maximizing adoption of the product platform by 3rd party developers. Internally, this unit became the champion

for 3rd party developers. Additionally the implementation of certification and API also implied new ways of working within the firm.

Reflection: In this case it was clear that the change trigger initiated from the ecosystem organizing level, in which the firm noticed changes in the way of interacting and collaborating with customers and developers. As is clear from this case, when a product reaches a customer base number in the hundreds of thousands and millions, there will be significant pressure by both customers and 3rd party developers, to 'open up the product' for customer and customer segment specific extensions. The patterns that we described in this case, is, we believe, quite generic for companies in this situation.

4.2 Case Beta: Pushed Back in Value Chain

Introduction to Case Beta. The case company Beta is a large global company in the embedded system domain. The unit that we studied works with OEM customers (Original Equipment Manufacturers) to provide one of the major sub-systems in their product. The company worked with the OEM's in the form of a solution provider, and delivered a dedicated subsystem implementation in response to the requirements from the OEM. The revenue of the company was generated by subsystem unit sales, where a subsystem unit consists of mechanical, hardware and software unit parts. Although the company provides software development services for its OEM customers, this was a negligible part of their business and received very little attention from general management.

Change Trigger. Until recently, the case study company Beta provided the complete solution to its OEM customers. Interviewees mentioned that some years ago, a shift started to occur in the unit's ecosystem. First, OEMs were starting to demand that software provided by the OEMs would be needed to be integrated in the overall solution, frequently replacing functionality developed by the case study company Beta. Second, OEM customers began to demand that the case study company provided arbitrary compositions of hardware, software and mechanics provided by competitors, the OEM and the case study company. For instance, in some cases, the case study company was requested to provide its software on hardware developed by competitors. In another case, the competitor software needed to be deployed on hardware developed by the case study company. The most complicated situations, however, were where OEM software, competitor software and software developed by the case study company needed to be integrated. The architectural boundaries in the three software subsystems did not align with each other, requiring deep integration and rework of already developed components to accomplish functional integration while achieving the necessary quality attributes.

During this shift the company's role in their ecosystem shifted from a turnkey solution provider to either a component provider or an integrator that worked under the close supervision of the OEM.

ESAO Analysis

Ecosystem Strategy: the role of the case company shifted from a turnkey solution provider to a component and integrator role. The case had to respond to what was happening in their ecosystem. The company explicitly designed the industry specific standardized architecture to coincide as much as possible with the sub system interfaces already existing in their platform architecture. This strategy would decrease the integration cost of customer and competitor subsystems that needed to be integrated into the company's products.

Ecosystem Architecture: the change in ecosystem strategy was driven by significantly increased integration costs, which severely impacted the profitability of the business. Due to this strategic change, the ecosystem architecture evolved into a much more modular architecture that allowed for replacement of subsystems of components provided by other parties.

Ecosystem Organizing: The case company used the existing standardization body to drive their standardization efforts. The case had to respond to the changes in their ecosystem by changing their role in the ecosystem. Their initial role was being a full solution provider but now the case study has shifted towards an integrator role and complementor role in their ecosystem.

Internal Strategy: the company went from an almost exclusively unit-based business model, towards a business model in which it split its units into three:

- (1) To a hardware-mechanical sales unit model that was based on the traditional sales unit model,
- (2) A software license sales and
- (3) Consulting service business where the organization would build and integrate software for any hardware mechanics configuration that the customer desired.

Internal Architecture: The internal architecture changed towards increased modularity. The original architecture was a highly integrated architecture towards optimizing hardware optimizing efficiency. This architecture was evolved into one into where modularity and decoupling between subsystems was prioritized at the expense of resource efficiency.

Internal Organizing: the changing business model and the increased architecture modularity caused the following changes in the organization:

- (1) For each type of business model a separate organizational unit was created. Especially in software this lead to a stronger separation between product platform development and customer projects.
- (2) In order to increase responsiveness to customers the interviewees mentioned that they had adopted agile work practices and are currently implementing continuous integration practices.

Reflection: In this case the changes in the ecosystem architecture with demands for integration triggered the other dimensions. As is clear from the description of the six dimensions of the ESAO model, there were several bi-directional interdependencies in case Beta, such as between the ecosystem and internal counterparts, as well as between the three aspects. For instance, although the organization identified that a shift in the ecosystem was taken place, no action was initiated until the integration costs for customer projects became unacceptably high. In other words, the Internal Organization initiated the change in all other dimensions, showing clearly that there is no sequential change process like BAPO, but a continuous alignment of the different dimensions.

4.3 Case Zeta: Strategic forward Integration

Introduction to Case Zeta. The case study company Zeta is a global company in the embedded systems industry. The industry in which case Zeta operates is highly fragmented with no company having more than 5% market share. The company has thousands of competitors, but it has been able to differentiate itself using product quality and reliability as key competencies. The products of the company have traditionally consisted of a major mechanical component and a minor hardware and software component. However, over the last decade, the R&D investment has shifted in a quite significant fashion towards software development. The company operates in a complicated technology ecosystem, consisting of wholesalers, retailers, installers, specification engineers, maintenance and end-customers.

Change Trigger. Over the last decade, many new competitors, especially in Asia (India and China), have appeared on the market, causing significant change in the ecosystem. Until recently, the original differentiator and niche of the company, i.e., quality and reliability, were sufficient to justify its market share and pricing power. During recent years, the quality of competitor products has reached a level that this differentiation strategy was no longer viable. After analyzing its strategic options, the company decided to forward integrate into its ecosystem and to start offering systems and solutions for which it originally only provided some of the components. The company started to change towards a turnkey provider with complex solutions in the HVAC industry (Heat, Ventilation and Air Conditioning). The reason for this is that the margins on systems and solutions were an order of magnitude higher than the margins of its traditional products.

ESAO Analysis

Ecosystem Strategy: as the company decided to forward integrate in its ecosystem, the company changed its role in the ecosystem from a component role towards a solution provider role. One of the main challenges of this change was to avoid upsetting its existing customer base. To accomplish this the company focused its new offerings initially on geographies where its primary customers have little or no market presence.

Ecosystem Architecture: the company originally provided its products as closed systems with minimal ability for system integrators and solution providers to access the product. Its ecosystem architecture strategy concentrated on two factors:

- (1) The adoption of industry standards to simplify the integration of its products as well as products from other manufacturers into systems and solutions.
- (2) It provided an ecosystem API that allowed third-party developers as well as its internal systems and solutions groups to extend basic product functionality with systems and solutions specific functionality.

Ecosystem Organizing: the company was looking to simplify the integration of products from other manufacturers into its own systems and solutions and had little incentives to market the capabilities to other players in its ecosystem. Consequently, it intentionally limited communication and interaction with others in the ecosystem.

Internal Strategy: although its product margins were under pressure, the company still was able to sell its products at a reasonable margin. Consequently, it adopted a dual business strategy.

- (1) On the one hand it sought to maximize the scale in its produce manufacturing and sales, seeking to drive down costs by maximizing scale.
- (2) On the other hand, it pro-actively build a number of system and solution business that, though high margin, were only able to provide limited unit sale levels.

Internal Architecture: the product architecture was optimized for minimizing hardware resource cost but allowed for extension with ‘apps’ through its ecosystem API. The tension in the organization was between minimizing hardware resource cost as desired by the product sales organization, and providing excess hardware resources in order to allow for system and solution specific apps to be installed on the device.

Internal Organizing: although the organization considered to create separate R&D departments for products and systems and solutions R&D, it instead developed a governance mechanism that allowed one R&D department to satisfy the hardware and software needs of the product systems and solutions business units. The governance mechanism consisted of a board representing all relevant stakeholders that met frequently, and prioritized the needs of products, systems and solutions.

Reflection: Based on the complex ecosystem strategy of the firm, and their role in the ecosystem, the company pro-actively decided to change its business strategy rather than being forced by other players in the ecosystem. However, again, this change had affect on all dimensions of the ESAO model. As we sought to highlight in this case, the importance is the alignment between the six dimensions, not which dimension initiates the change.

5 Conclusion

Software ecosystems and open innovation are increasingly important for companies as well as market segments. Current ecosystem research primarily looks at the ecosystem – but does not link this back towards the internal organization or to the implications of the internal organization, software platform or architecture and the ways of working. This is a challenge for many organizations for several reasons. First, the way the organization works internally and the way the company engages with its ecosystem need to be closely aligned with each other, as a strong co-dependency exists between the two. Second, as companies increasingly seek to focus their internal efforts on what truly differentiates them and try to outsource as much as possible of the non-differentiating activities, the reliance on the ecosystem is increasing significantly. Finally, as customers are transitioning from buying products to acquiring services, the burden of integrating different products into a dedicated solution for customers is falling to ecosystem players that need intimate interactions with other players in the ecosystem. However, few, if any, holistic models exist that support organizations to better align their internal and ecosystem dimensions.

In this paper we introduced the EASO model that encompasses elements of both the ecosystem as well as the internal organization. The model provides three perspectives, i.e. strategy, architecture and organizing, and two dimensions, i.e. internal and ecosystem. This provides an approach where the alignment between the perspectives and between the internal and ecosystem dimensions can be analyzed and improved. The development of a new analysis and assessment model including not only the internal organization, but also an external ecosystem dimension is an important improvement of earlier models that primarily focused on either the internal company, e.g. the BAPO model [13], or that only focus on one of the dimensions like the ecosystem, the architecture, or way of organizing.

Through our case examples we show that the new ESAO model is able to incorporate the external and internal strategy, architecture and ways of organizing. Furthermore, the model is applicable not only for product-line companies, but also for other companies that work with software intensive and embedded systems. Finally, the ESAO model is not a sequential model, in which one dimension is followed by another dimension. Instead, the ESAO model focuses on analyzing and achieving alignment between all the different dimensions. The different dimensions of ESAO impact each other and need to be aligned for software intensive organizations in order to be able to develop their organization within their particular ecosystem as well as to react to changes occurring in their ecosystem.

From the case examples it also became clear that firms with different roles in their ecosystem choose different strategies to either maintain their role or re-act to changes in the ecosystem and ecosystem roles. However, the different roles played by firms might have an impact on the alignment of the ESAO components. It might be that different ecosystem roles imply a trade off or prioritization of the other ESAO elements since these different ecosystem roles might have different implications for the architecture, strategy and way of organizing within a firm. Future work would

need to study the implications of working in multiple ecosystems with different roles for the internal and external dimensions.

The proposed model of ESAO offers an analysis and assessment framework, as well as a framework supporting change and development within software intensive organizations. In this paper, the focus is on the analysis, but in future work, we intend to expand on the other uses for the model as well.

References

1. Moore, J.F.: Predators and prey: a new ecology of competition. *Harvard Business Rev.* 71(3), 75–86 (1993)
2. Moore, J.F.: Business ecosystems and the view from the firm. *The Antitrust Bulletin* 51(1), 31–75 (2006)
3. Iansiti, M., Levien, R.: Strategy as ecology. *Harvard Business Rev.* 82(9), 69–78 (2004)
4. Gawer, A., Cusumano, M.: *Platform leadership*. Harvard Business School Press, Harvard (2002)
5. Gawer, A., Henderson, R.: Platform owner entry and innovation in complementary markets: evidence from Intel. *J. of Eco. and Management Strategy* 16(1), 1–34 (2007)
6. Li, Y.-R.: The technological roadmap of Cisco's business ecosystem. *Technovation* 29, 379–386 (2009)
7. Bosch, J., Bosch-Sijtsema, P.M.: From integration to composition: On the impact of software product lines, global development and ecosystems. *J. of Systems and Software* 83, 67–76 (2010)
8. Manikas, K., Hansen, K.M.: Software Ecosystems –A Systematic Literature Review. *J. of Systems and Software* 86(5), 1294–1306 (2013)
9. Hofmeister, C., Kruchten, P., Nord, R.L., Obbink, H., Ran, A., America, P.: A general model of software architecture design derived from five industrial approaches. *J. of Systems and Software* 80(1), 106–126 (2007)
10. Bosch, J., Bosch-Sijtsema, P.: Coordination between global agile teams: From process to architecture. In: Šmite, D., Brede Moe, N., Ågerfalk, P. (eds.) *Agility Across Time and Space. Implementing Agile Methods in Global Software Projects*, pp. 217–233. Springer, Berlin (2010)
11. Larman, C.: *Agile and Iterative Development: A Manager's Guide*. Addison-Wesley, Boston (2004)
12. Schwaber, K.: *Agile Software Development with Scrum*. Prentice Hall (2001)
13. van der Linden, F.J., Dannenberg, R.B., Kamsties, E., Käsälä, K., Obbink, H.: Software product family evaluation. In: Nord, R.L. (ed.) *SPLC 2004. LNCS*, vol. 3154, pp. 110–129. Springer, Heidelberg (2004)
14. Teece, D.J.: Business models, business strategy and innovation. *Long Range Planning* 43, 172–194 (2010)
15. Boudreau, K.J., Lakhani, K.R.: How to manage outside innovation. *MIT Sloan Management Rev.* 50(4), 69–76 (2009)
16. Olsson, H.H., Alahyari, H., Bosch, J.: Climbing the “Stairway to Heaven”—A Multiple-Case Study Exploring Barriers in the Transition from Agile Development towards Continuous Deployment of Software. In: *38th EUROMICRO Conference on Software Engineering and Advanced Applications (SEAA)*, pp. 392–399. IEEE Press (2012)
17. Bosch, J.: Achieving Simplicity with the Three Layer Product Model. *IEEE Comp.* 46(11), 34–39 (2013)