A Survey of Associate Models used within Large Software Ecosystems

Joey van Angeren[1], Jaap Kabbedijk[1], Slinger Jansen[1], and Karl Michael Popp[2]

[1] Department of Information and Computing Sciences, Utrecht University
Princetonplein 5, 3508 TB Utrecht, the Netherlands
{j.vanangeren,j.kabbedijk,s.jansen}@cs.uu.nl
[2] SAP AG, Corporate Development
Dietmar-Hop-Allee 16, 69190 Walldorf, Germany
karl.michael.popp@sap.com

Abstract. Associate models are powerful tools for large software ecosystem orchestrators to manage clusters within its ecosystem. At present, however, it remains unclear how these associate models are used in practice and of what elements such models consist. Without an overview of what associate models consist of, the concept of software ecosystem orchestration will remain elusive. In this paper, a conceptual overview is presented that describes the structure of an ecosystem associate model. The conceptual overview consists of the roles fulfilled by the participant within the associate commitment, including the dimension of the role as well as resulting benefits, requirements and costs. Furthermore, the conceptual overview enables a categorization into different forms of associate model governance, entry barriers and goals for the three respective models. With the conceptual overview, software ecosystem orchestrators can develop their own associate models and attain insight into the forces that are at play in their own software ecosystem.

Key words: software ecosystems, partner ecosystems, associate models, partnership characteristics, partnership models, membership models

1 Introduction

Software vendors and platform owners act within a network of actors relevant to their business, called a software ecosystem. A software ecosystem is defined as "a set of actors functioning as a unit and interacting with a shared market for software and services, together with the relationships among them"[8]. A software vendor's software ecosystem may consist of multiple subsystems, for example, a supplier ecosystem that contains all the suppliers and a partner ecosystem [12]. Different roles performed by actors in an ecosystem are, for instance, sales partners, system integrators, and value added resellers. By looking at these roles, it becomes possible to identify clusters or software ecosystem subsystems, further referred to in this paper as associate models. This paper discusses the identifying characteristics of the commitments that define these models.

Software ecosystem orchestrators frequently choose to engage into a set of partnerships in the search for mutual benefits. Some large software ecosystem orchestrators expand this principle to a larger scale by creating a cluster of organizations around themselves. Clusters, in this sense, are a number of actors that are grouped together. Organizations, such as software vendors, platform owners and open source associations use associate models to manage these clusters of participants in their ecosystems. In this paper, two of these methods are discussed in the form of partnership models and membership models.

A partnership model is owned by a software vendor or platform owner to sustain, manage, cluster and expand their partner ecosystem and therefore the number of actors within this ecosystem. A partnership model offers organizations the possibility to engage into a partnership with a major software vendor or platform owner. In this model, the partner fulfills one or more roles that brings a certain set of predefined benefits. To be able to obtain or retain a role, the partner has to comply to a certain set of requirements and possibly has to pay fees to the cluster owner. Open source associations and platform owners offer a similar model to sustain, manage, cluster and expand their software ecosystem. Because of the different legal form of those organizations, actors within these clusters are members that are part of the membership model. Partnership and membership models are examples of big clusters of participants within one or more areas of the software ecosystem, or a subsystem of this ecosystem, referred to as associate models.

The emphasis of this paper is on the structure of associate models within the software industry and the identifying characteristics of commitments within those models. Few previous research exists within this specific part of the software ecosystems research area. Understanding the structure of these associate models will improve the understanding of the implications of owning or participating in such a cluster can have. Also, the presented results are valuable for software ecosystem orchestrators when setting up their own affilate model. The results presented in this paper are formulated around a conceptual overview that captures the structure of an associate model. Three case studies of large software ecosystem orchestrators owning an associate model, that are representative for those within the industry, provide support for the findings presented in this paper. The cases will also be used to identify the main differences between a partnership and a membership model through classification.

This paper continues with a description of the research approach in section two, in which we will elaborate on the research methods we employed as well as the data collecting process. In the third section, we present the result of a literature review, which gives an overview of what is already known about associate models. The created conceptual overview that captures the structure of associate models is presented in section five. Section six provides an overview of the gathered data out of the three case studies, followed by a classification of these three cases in chapter seven. In section eight we draw the draw conclusions and provide suggestions for future research.

2 Research Approach

The main research question of this article is as follows; *"What are the identifying characteristics of a commitment within an associate model?"* To be able to answer this research question, a combination of literature review and case studies is carried out. The gathered data from literature review and documentation, large software ecosystem orchestrators provide on their cluster (e.g. website, contracts), of multiple associate models form the basis for a conceptual overview. This conceptual overview is created to describe the general structure of an ecosystem participant cluster. The conceptual overview is created by applying design science [4], a cycle that is based on a continuous process of developing, assessing evaluations and refining the created deliverable. After constructing the conceptual overview according to findings from literature and documentation, expert reviews and case studies were employed to evaluate the constructed conceptual overview, according to which modifications have been made. Furthermore, this conceptual overview forms the basis for a large number of characteristics identified and used for a classification of multiple partnership and membership models that exist within the industry.

Associate models, owned by three large software ecosystems, that are representative for those owned within the software industry, are used as cases for this paper. Each case study did start out with studying available documentation on the respective ecosystem participant cluster (e.g. website, contracts). Later, a semi-structured interview has been conducted with a representative of each of the cluster owners. For these semi-structured interviews we employed an interview protocol to provide guidance during the session. The three models are compared to identify the main similarities and differences between them. We chose for a multiple case design [13] approach to provide a diverse set of results.

3 Associate Models

Software ecosystems exist around one particular software vendor or platform owner. Jansen et al. [7] define multiple types of software ecosystems by the way in which they are centered around one or more specific actors, for example; market, technology, platform and firm software ecosystems. Out of these four types of software ecosystems, the platform software ecosystem and firm software ecosystem are relevant for this paper. In case the software ecosystem forms around a platform, the platform owner becomes the keystone [6] in the software ecosystem. On the other hand, a software ecosystem can exist around one particular software vendor, for example the Microsoft or SAP software ecosystem. One component of this software ecosystem is the partner ecosystem, where various partners are clustered.

A software vendor can have a wide range of goals with its ecosystem. Popp [11] defines in a simplified setting five different types of goals a software vendor or platform owner might have; financial, customer related, product related, network related and market related goals. Since the partnership ecosystem is a component of the software ecosystem, those goals also apply for the partner ecosystem. One or more of these goals can be achieved through the use of an ecosystem participant cluster and are formulated within the business model. Most prior research on partnership ecosystem goals has emphasized on product and market related goals. Bosch [1] for example, elaborates in detail on ways to achieve product related goals, by making partners co-innovate products from the software vendor, by taking up the role of a value-added reseller or by letting partners develop further functionality on top of the software vendor's platform or product. Cusumano [3] opts for a similar approach to reach product related goals and advises companies to use partnering as a tool to extend their market reach. While partnering is about the search for mutual (strategic) advantages, little research exists on partnering out of a software vendor's or system integrator's perspective. Neither for research that elaborates on the benefits, disadvantages and risks of fulfilling one or more roles in a partnership model.

Organizing a large amount of participants around an organization in a cluster is a practice that happens not only within the software industry. The clustering of participants is also a common phenomenon in other business areas, like the automotive industry [10, 9, 2]. While the principle of clustering participants around an organization is similar, as well as a certain amount of co-innovation opportunities, in this industry the clustering of participants is mainly focussed on the (cooperative) buyer-supplier or manufacturer-distributor relationship.

4 A conceptual overview for Associate Models

Because of the character of an ecosystem participant cluster, it requires a governance structure to keep it manageable. The ecosystem participant cluster consists out of multiple levels or roles in which the participants are clustered. Within this cluster, every participant has one or more roles with predefined requirements, benefits and costs. While in practice multiple associate models from different software vendors, platform owners or open source associations differ from each other, the main principles behind it are the same. Based on this, the core structure of the different models used in practice are similar or consist of the same core elements. With this observation and to illustrate this, a conceptual overview is created to give a visual representation of the typical structure of an associate model owned and utilized within the software industry. The model is created on a type level, in Unified Modelling Language using modelling techniques similar to those used in [5].

Figure 1 contains a conceptual overview that describes the structure of an associate model within the software industry. Such a model is described by a name, a description and an owner. Every associate model can be classified by a model type. The type that classifies the model influences the primary structure and restricts the commitment possibilities that exist within the offered model. The associate model consists of partnerships or memberships with organizations wishing to engage in them.

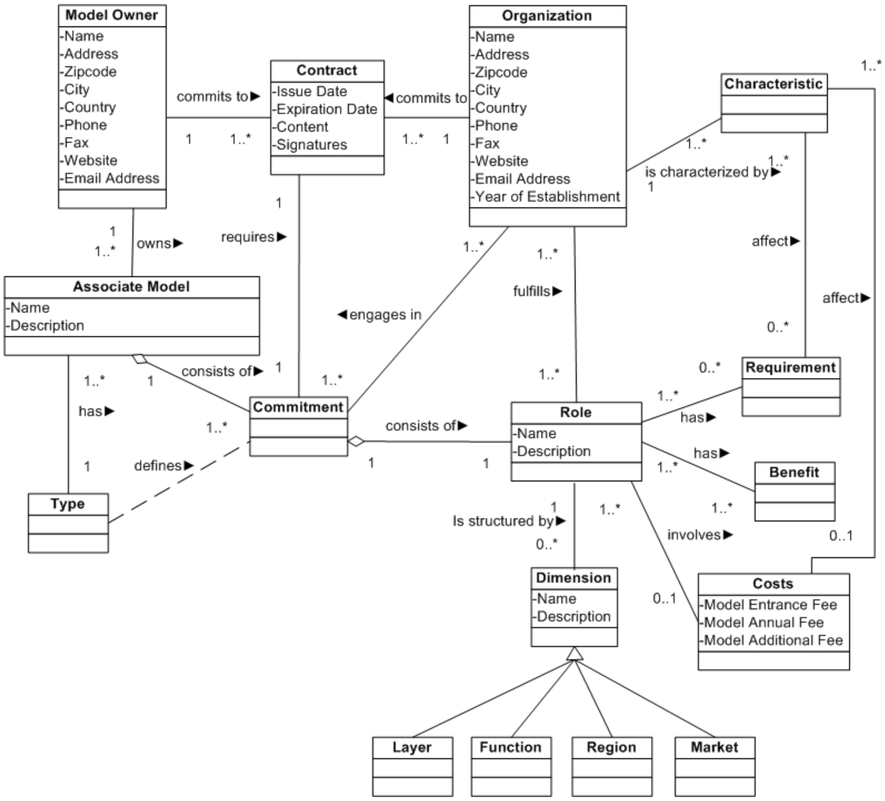

Fig. 1. A conceptual overview of an associate model

An organization can engage in one or multiple commitments with the model owner in which it fulfills a role. Each partnership or membership has a unique contract which is a commitment between two legal entities, the model owner and the organization that engages in the commitment. In practice this means that, for every role an organization fulfills within an associate model, a separate contract will be signed by both legal entities. A role within the associate model is characterized by a name and a description and can have zero, one or more dimensions. This can be best explained by an example. A software vendor owns a partnership model that is hierarchical, the primary structure consists of three levels (i.e. bronze, silver and gold). Within these levels there is a next level of decomposition, a horizontal one in which partners are organized by their respective functions (e.g. system integrator, value-added reseller). This means there are partners that fulfill a role as a bronze level system integrator and other partners fulfill the role of a gold level system integrator.

Furthermore, every single role has a set of predefined benefits for an organization. On average, a wide range of benefits is offered or expected, ranging from marketing benefits or coaching to getting access to the source code of software products for the highest level partners or members within some models. To receive those benefits, a set of requirements have to be met by the participant in order to be eligible to fulfill or to retain a certain role. The requirements that an organization has to meet may be defined by some of the specific organizational characteristics of the participant. Geographic location or total corporate revenue for example might influence the requirements to be met. Furthermore, requirements for a system integrator might differ from those of a value-added reseller even if they have the same level or role within the model.

An organization that is willing to become part of a cluster within the partner or software ecosystem pays to be eligible to fulfill a certain role. Exceptions to this are educational and community memberships, in which participants get small benefit from or contribute to the community of organizations that are active within the parts of the ecosystem that are targeted with the associate model. Usually, an annual fee has to be paid at the contract renewal date, however, model entrance fees as well as additional fees might be applicable. In some cases, specific characteristics of the organization might affect the costs for engaging in a partnership or membership. Companies with different geographic locations or a different total corporate revenue might pay different fees, as stated in predefined tables or scales.

5 Case Studies

Three large software ecosystem orchestrators, each possessing their own associate model, that are representative for those within the software industry, have been selected as case studies for this paper. Each case study started out with studying the documentation on the respective model that is publicly available (e.g. website, contracts). With the gathered knowledge out of documentation, the conceptual overview and the literature review as a basis, a semi-structured interview was conducted with an expert from each of the three organizations to gather additional information with regard to their associate model. For this, we utilized an interview protocol to provide guidance during the session. The topics of interest within these case studies include; model structure, benefits, requirements, entry barriers and goals strived for by the model owner. The case studies are used to further evaluate the conceptual overview as well as to gather additional information about the three respective models.

5.1 Case Study: SAP

SAP is a major player within the global enterprise application software market with their platform, products and services. The partner ecosystem is a substantial component of the SAP software ecosystem and is managed by their own partnership models. Those include a global partnership model and several region-specific partnership models. The global partnership model is constructed around ten different function-based roles targeting specific type of partners, like service providers, software partners and value-added resellers. Within some of these roles dividing partners further into market- or function-based categories is possible, this is however not regarded as a second dimension within the model. Furthermore, the model contains one, more general, umbrella role; the SAP PowerEdge Partner Program. In the SAP partnership model, partners can fulfill more than one function-specific role. The partnership model is documented on different parts of the SAP website. Internal documents and contracts are not publicly accessible.

Each role has a specific set of benefits, requirements and costs. Upon request small changes to these sets can be made. Benefits and requirements per role are targeted on specific needs for the type of partner that fulfills a role. Requirements involve completing a certification process for services, tools or software solutions that are somehow related to SAP or the SAP platform. Benefits range from marketing benefits to revenue sharing agreements with SAP, SAP incorporating a solution within their products or getting access to the SAP customer ecosystem, to sell products or to offer additional services. The partnership model is utilized by SAP to reach various goals, ranging from extension of the partner ecosystem, product or platform development and co-innovation to monetizing on the partner ecosystem and extending market reach. SAP actively monitors the performance of their partners, so that good performing partners can progress from certification partners to, for example, resellers.

Apart from the annual partnership fee, the certification process can be regarded as a main entry barrier for the model. A partner has to devote resources to create a product that is eligible to successfully pass the certification process. Furthermore, potential partners are screened on their involvement in or with other platforms. This means that a partnership with SAP becomes a main part of the business, but on the other hand that also means a partner can lock himself out of participating in another partnership model within the industry that has a similar policy.

5.2 Case Study: Open Design Alliance

The Open Design Alliance is best described as an association of developers. The nonprofit organization develops the Teigha platform, a platform for CAD and other technical graphics applications. ODA is a member-driven organization and their membership model, that consists of over 1200 members, is the core of the organization. Organizations that want to obtain or use the Teigha platform have to enter the membership model. The membership model of the ODA can be best described as a layered model without a second level of decomposition. The model consists of five unique levels. The main benefit of each level is targeted at the amount of access members have to the Teigha platform. Ranging from the in-house use of the platform to access to obtaining access to the source code and no restrictions in the way the platform is used or incorporated. Additional benefits for each level are mainly targeted at the amount of influence a member has on the development of the platform or the way in which it wishes to contribute to the development. Each organization that wants to become part of the model can only apply for one membership level and customizations to these levels upon request will not be made.

Apart from the educational membership, which is designed to stimulate research, every membership level requires an annual membership fee to be paid. When a new member desires to enter the membership model it has to pay a one time only membership model entrance fee. The additional entrance fee can be regarded as a raised entry barrier for the membership model, the Open Design Alliance did however opt for this approach to stimulate members to engage into long-term commitments. By making all the documentation on the membership model publicly available, the ODA aims to make the membership model and its principles as accessible as possible.

The main goal the ODA has with their membership model is the expansion of the Open Design Alliance software ecosystem. Another goal ODA has with their membership model is the development of their platform, a product related goal. The majority of membership model fees flow back into development and in cooperation with members new features or additional functionality can be incorporated into the platform. The ODA aims for a certain level of empowerment between members and platform owner as well as creating a supportive community around the platform.

5.3 Case Study: Eclipse Foundation

The Eclipse Foundation is a member supported corporation that provides a platform, tools and services for the Eclipse software ecosystem. There are multiple ways for organizations to contribute to the development of one or more Eclipse platforms or solutions. They can make donations (financial or resources), can become a corporate sponsor or can become a more prominent part of the Eclipse software ecosystem by becoming a member. If an organization wishes to become a member it will take part in the Eclipse Membership Program.

The Eclipse Membership Program consists of five different levels. The membership model is based on the open source maturity curve. This maturity curve predicts the way in which an organization intensifies their involvement in an open source community over time as it evolves and develops as an organization. Because of this, every next membership level can be considered as superior to the previous one. Therefore, it is highly unlikely that an organization will occupy more than one membership level. Predefined membership levels will not be customized upon request of a potential member, they can only choose to not be publicly listed as a member.

Benefits per membership level range from contributing to the platform or Eclipse solutions, to influence in the overall governance of the Eclipse Foundation, since Strategic members get one seat within the board of directors. Furthermore, higher level members can benefit from industry specific working groups in which fellow members cooperate and co-innovate the Eclipse platform to suit industry specific needs. To maintain or occupy a higher membership level, an annual fee has to be paid. For most membership levels additional requirements have to be met. Organizations will, for example, have to release a solution, with incorporation or extension of the Eclipse platform, within twelve months after becoming a member, or will have to devote a certain number of full-time developers to the development of the Eclipse platform or tools. To lower entry barriers for certain levels, requirements or annual membership fees are adjusted to specific organizational characteristics like the total number of employees or total corporate revenue. The main goals the Eclipse Foundation has with their membership model are further expanding the Eclipse software ecosystem, the development of the Eclipse Foundation platform and tools as well as getting a higher level of involvement from actors within the software ecosystem.

6 A Classification of Associate Models

To compare the three associate models, as presented in the case studies, a classification table has been created. A number of characteristics for associate models has been derived from the conceptual overview, the interview protocol and the content of the case studies as well as the software ecosystem goals described by Popp. [11]. Those have been grouped into relevant categories. The characteristics are used to identify the main similarities and differences between the three respective models.

Table 1 contains a classification of the three studied associate models. It gives an overview of the different characteristics of each of the models, ordered within different categories. As became clear in the case studies, the structure of different associate models can be described by one conceptual overview, regardless the type of platform offered (e.g. open or closed source), the type of the model (e.g. partnership or membership model) or the primary structure of the cluster.

Both open source platform owners posses an associate model in the form of a membership model, while SAP owns a partnership model. The membership models have a layered primary structure that is based on increasing involvement of organizations in their contribution to open source projects. This is most explicit in the Eclipse model, which is entirely constructed based on an open source maturity curve. While the layered models are based on contribution to the platform, SAP manages their model out of a business performance point of view. By monitoring the performance of their partners they can choose to up-scale the partnership agreements from a certified partnership to a reseller agreement. This possibility should enrich benefits for both parties in the form of, for example, sharing revenues. Because of this, partners do not progress through levels or roles, they rather evolve within their existing role.

The other differences in characteristics between the three associate models are caused by the differences in primary structure between the respective clusters. The function-based partnership model has a richer set of roles because it has to suit a wide range of business needs, for different kinds of organizations as well as for themselves. For this reason, partners can fulfill more than one role in a partnership model when they provide a broad range of services. In the Eclipse membership model this is theoretically possible as well. This, however, will not happen because higher levels are regarded as superior to previous ones.

Table 1. A classification of associate models

Category	Characteristic	SAP	ODA	Eclipse
Platform	Open/Closed source	Closed	Open	Open
Structure	Layered primary structure	N	Y	Y
	Role-based primary structure	Y	N	N
	Market-based primary structure	N	N	N
	Number of dimensions	1	1	1
	Total number of roles/levels	11	5	5
	More than one role/level can be fulfilled by the same organization	Y	N	Y
	Dependency between organizational characteristics and requirements	N	N	N
	Dependency between organizational characteristics and costs	N	N	Y
Entry Barriers	Model has annual fees	Y	Y	Y
	Model has one time only entrance fees	N	Y	N
	Model has roles/levels free of charge	N	Y	Y
	Devoting resources is regarded as an entry barrier	Y	Y	Y
Model Governance	Governance includes platform defence	Y	N	N
	Roles/Levels are customizable upon request	Y	N	N
Documentation	Model is documented on website	Y	Y	Y
	Contracts are openly accessible	N	Y	Y
Goals	Utilized to achieve financial ecosystem goals	Y	Y	Y
	Utilized to achieve customer related ecosystem goals	Y	Y	Y
	Utilized to achieve product related ecosystem goals	Y	Y	Y
	Utilized to achieve network related ecosystem goals	Y	Y	Y
	Utilized to achieve market related ecosystem goals	Y	N	N

One of the important categories of characteristics, besides the structure of the model, are the identified entry barriers. Each role or level a participant fulfills or occupies within an associate model comes with certain costs. All model owners charge the participants acting within their model with an annual fee, except for the educational and community roles or memberships that are free of charge. In addition to the annual fees, ODA includes one time only membership entrance fees into their model, to both protect their intellectual property and to stimulate members to opt for a long-term commitment.

Furthermore, contributing resources is regarded as a main entry barrier. In both membership models resources will have to be contributed to product and platform development. In the Eclipse model, for example, it can be required for a participant to contribute a number of full-time developers to platform development. The same in a different form goes for the SAP partnership model. To be able to become a partner, an organization has to certify their product or service so that they comply with guidelines provided by SAP. Organizations will have to devote resources in order to do so. Devoting resources to co-innovation with SAP is also a possibility. An additional entry barrier for the SAP partnership model is a result of their model governance. Organizations can lock themselves out of the model by having a high level of involvement with another large software ecosystem (for example as a reseller), while a lot of involvement with SAP can lock them out of models offered by competitors of SAP.

All three software ecosystems aim for a wide range of goals with their ecosystem participant cluster. For all three organizations, network related goals are a main reason to own an ecosystem participant cluster, to utilize this cluster to expand their software or partner ecosystem and therefore the number of actors within this ecosystem. For ODA, their membership model is the core of their business, which makes the model even more prominent to achieve goals. For the membership models, the main goals are product related goals. Both ODA and Eclipse strive to utilize their membership model to develop and innovate their platform. For ODA membership fees flow back into development, while members also actively contribute, by for example, creating new functionality on top of the existing platform. Higher level Eclipse Foundation members contribute full-time resources to the development of the Eclipse platform. In the SAP partnership model, product related goals are less prominent, however, co-innovating with partners to strengthen SAP offerings is a goal strived for.

Financial motives for Eclipse and ODA do exist, because of the membership fees, for SAP those are more prominent. They monetize on their partnership model by receiving annual fees, but apart from that they strive to create additional revenue streams. Those revenue streams can be a direct result from partner performances within a certain role but can apart from that also be created by reseller or revenue sharing agreements. Market related goals are most relevant for SAP as well. SAP utilizes their partnership model in an attempt to increase market presence through the ecosystem of their partners. Furthermore, they can enter markets leveraging their partners.

The main reason for the identified differences comes with the different legal form, the type of platform and the business model of the three respective organizations. As a consequence, the associate models have a different position and target area within the organization and parts of the ecosystem. SAP is a profit organization, while both ODA and Eclipse are a non-profit association with members. As a consequence, the membership models offers a set benefits that covers a wider range. Benefits are not limited to platform, financial or strategic related benefits. Rather benefits stretch a wide range, including influence in the actual decision making process within the association. Because of the difference in organizational structure, ODA and Eclipse membership levels are not customizable, while the roles in the SAP partnership models are. Open source associations are slim organizations that aim to reduce administrative pressure by containing a rigid governance regarding customizations.

7 Conclusion and Discussion

In this paper, we addressed two ways in which a large software ecosystem orchestrator can manage clusters within its ecosystems, through partnership and membership models. These models can be utilized to achieve different goals, depending on their business model. associate models consist of a set of commitments between model owner and participant. Although partnership and membership models differ from each other, their overall structure is the same. By applying design science, a conceptual overview has been created to capture the structure of associate models. In every commitment within such a model, a participant fulfills one or more roles with a predefined set of benefits, requirements and costs. Roles have zero, one or more dimensions enabling participants to engage into multiple levels of commitments with the model owner. Although the conceptual overview has been created based on multiple partnership and membership models, not limited to the three case studies presented in this paper, more research is needed to further evaluate and validate the conceptual overview.

The created conceptual overview, the interview protocol and findings from the literature review did provide the basis for multiple associate model characteristics, that form the basis for a classification. Since the characteristics have been selected independently from the actual content of the case studies these characteristics can be used to classify other associate models utilized within the industry. Differences in structure between the three studied associate models are identified, the membership models are layered, while the partnership model is role-based. The way in which participants develop within the model overtime differs. Within the membership models, participants progress through levels, while in the partnership model participants evolve within their existing role. Financial motives, the devotion of resources and a high level of involvement with another platform, are main entry barriers for the associate model that can keep potential participants from participating. Furthermore, the way in which large software ecosystem orchestrators strive to achieve goals with their associate models differs. The generalizability of identified differences between partnership and membership models is limited because of the small number of case studies employed and because the information does not come from an independent source. However, the majority of differences identified are a result of organizations differing from each other in legal form, different business models and the type of platform they posses. Strategical decisions influenced by these factors reflect back in the characteristics of the associate model and the set of commitments it consists of.

Although previous research exists on associate models, especially partnership models, little research exists out of a participants' perspective. Since participating in an associate model affects or changes the business model of a participant, further research has to be edged on advantages, risks perceived as well as disadvantages, resulting from participating within an associate model. This also includes goals and expectations, organizations have with participating within such a cluster. Furthermore, studying the community-effect within these associate models out of a model owner' perspective may help improving the collaboration between actors within this cluster.

References

1. J. Bosch. From software product lines to software ecosystems. In *Proceedings of the 13th International Software Product Line Conference*, pages 1–10, 2009.
2. S. Chen and W. Wu. A systematic procedure to evaluate an automobile manufacturer-distributor partnership. *European Journal of Operational Research*, 205(3):687 – 698, 2010.
3. M. A. Cusumano. *The Business of Software: What Every Manager, Programmer, and Entrepreneur Must Know to Thrive and Survive in Good Times and Bad.* Free Press, 2004.
4. A.R. Hevner, S.T. March, J. Park, and S. Ram. Design science in information systems research. *MIS Quarterly*, 28:75–105, 2004.
5. I. Hunink, R. van Erk, S. Jansen, and S. Brinkkemper. Industry taxonomy engineering: the case of the european software ecosystem. In *Proceedings of the Fourth European Conference on Software Architecture: Companion Volume*, pages 111–118, 2010.
6. M. Iansiti and R. Levien. *The Keystone Advantage: What the New Dynamics of Business Ecosystems Mean for Strategy, Innovation, and Sustainability.* Harvard Business School Press, 2004.
7. S. Jansen, S. Brinkkemper, and A. Finkelstein. Business network management as a survival strategy: A tale of two software ecosystems. *1st International Workshop on Software Ecosystems*, 505:34–48, 2009.
8. S. Jansen, A. Finkelstein, and S. Brinkkemper. A sense of community: A research agenda for software ecosystems. In *ICSE09: Proceedings of the 31st ICSE Conference on Software Engineering*, pages 187–190, 2009.
9. K. Langfield-Smith and M. R. Greenwood. Developing co-operative buyersupplier relationships: A case study of toyota. *Journal of Management Studies*, 35:331–353, 1998.
10. S. Nazli Wasti, M. Kamil Kozan, and A. Kuman. Buyer-supplier relationships in the turkish automotive industry. *International Journal of Operations & Production Management*, 26:947 – 970, 2006.
11. K.M. Popp. Goals of software vendors for partner ecosystems a practitioners view. In *Software Business*, volume 51 of *Lecture Notes in Business Information Processing*, pages 181–186. Springer Berlin Heidelberg, 2010.
12. K.M. Popp and R. Meyer. *Profit from Software Ecosystems: Professional Edition, Business Models, Ecosystems and Partnerships in the Software Industrys.* Books on Demand GmbH, 2010.
13. R. K. Yin. *Case Study Research: Design and Methods.* Sage Publications, Inc, 2008.