# The Orchestrator's Partner Management Framework for Software Ecosystems

Paul van Vulpen, Slinger Jansen, Sjaak Brinkkemper

*Utrecht University, the Netherlands*

**Abstract**

In software ecosystems, partner management concerns establishing and maintaining meaningful relationships with partners to improve ecosystem health. At the same time, software ecosystem orchestrators have insufficient insight into partner management tools. Because of this, they fail in attracting and maintaining partnerships with software producing organizations, ultimately hindering their ability to build healthy software ecosystems.

This research presents a classification for partner management activities to aid orchestrators in establishing and maintaining partnerships. We contribute (1) a methodology to assess partner management in software ecosystems, (2) a classification of knowledge and activities for partner management, and (3) a starting point for the development of theory regarding partner management activities. The classification and the activities are evaluated in five case studies of software platform orchestrators.

*Keywords:* Software Ecosystem Health, Partner Management, Software Ecosystem Orchestration, Software Ecosystem Governance

## 1. Introduction

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. These relationships are frequently underpinned by a common technological platform or market and operate through the exchange of information, resources, and artifacts [43]. Software ecosystems have emerged as a concept to characterize and understand modern collaboration between Software Producing Organizations (SPOs) in the software industry. In the software industry orchestrators have emerged; organizations that take responsibility for managing interactions between actors within the software ecosystem [47] and purposefully build and manage inter-firm networks [56,29,21]. Network orchestration is the process of assembling and managing an inter-organizational network to achieve a collective goal [49], in which the role is accepted by the other network members [48]. Orchestrators that are able to balance their own interests by bringing joint benefits for other players are likely to create healthy software ecosystems [2].

SPOs are legal entities in which the production of software is the main business activity and aim for their software to be promoted and used [45]. Typically, these SPOs have as their main goal to optimize profit. SPOs will probably never be able to operate independently: they rely on technology stacks, developed by other SPOs, to create profitable technologies for customers. These technology stacks are run by software ecosystem orchestrators. To use the technology from orchestrators, SPOs must generally become, although sometimes begrudgingly, partners of the orchestrator, established in a formal partner agreement. One of the challenges for both SPOs and orchestrators is that SPOs want to extract value from customers, while orchestrators want to extract value from the relationship with, or the works of, the SPO. This strains the relationship and requires coordinated management. Finally, orchestrators compete with rivaling platforms who also intend to form partnerships with the SPOs and even offer them opportunities for multi-homing [34].

Managing a software ecosystem comes with various responsibilities and roles. It requires overseeing development of parties and management [56], maintaining relations with the actors [35,42,71], coordination of actor relationships [11], stimulating innovation [44], while also purposefully building and managing the

inter-firm network [56,29,21].It requires the assembly and management of an inter-organizational network to achieve a collective goal [49], achieving the commitment and trust from other network members [48], and creating the resources that contribute to other members' business goals, also known as ecosystem enablers [L5]. The scope of this research is limited to partners that develop and monetize platform extensions, and we exclude consulting partners, sales partners, or infrastructure providers.

Software ecosystems can be considered healthy when they are active, produce valuable and usable software, manage to retain and attract users and partners, and possibly create revenue for the SPOs in it [41]. Successful orchestrators are able to create healthy software ecosystems [39]. These orchestrators must also balance health of the software ecosystem with their own sustenance; orchestrators also need a business model to stay in existence [51]. In this work, we equate platform owners and orchestrators, even though examples exist where the orchestrator is not the platform owner. Iansiti & Levien propose that ecosystem health should be measured by productivity, robustness, and niche creation [35].

Productivity can be measured in return on invested capital; how much value is created turning tangible and intangible assets into production. One ecosystem enabler is the creation of an architecture that is extendable by partners. By introducing boundary resources, defined as the key means for exposing and extending the platform architecture in order to facilitate third-party application development [26], partners can rapidly extend platforms with valuable complementary features [27,39,60]. Robustness is measured in survival rate of ecosystem's members, either in relation to other ecosystems or over time. An example of an ecosystem enabler that strengthens robustness is the investment of the orchestrator into joint sales and marketing initiatives, generating revenues for both the partner and the orchestrator. Niche creation refers to the ability to create value by putting new functions into operation and increasing meaningful diversity in ecosystem through that. An example of an ecosystem enabler that supports niche creation is the investment into startups within the software ecosystem. Intel, for instance, has made available one billion dollars for investment into new startups [1], ensuring that new niches will be created within the software ecosystem.

Creating a healthy software ecosystem requires the collaboration with partners, as it comes with numerous benefits [25]. Orchestrators, however, struggle to manage partners and have little insight in the governance tools that are available to them [40]. While partner management can be the dominant competitive advantage of platform leaders [61,62], orchestrators struggle with more elementary topics such as balancing value creation and realization within ecosystems [L4], and establishing partner relations [L13].

The partnering process of SECO orchestrators can be seen from three perspectives. First, the domain of partner acquisition addresses how partners should attracted, selected, and engaged and these topics are addressed in a sister article to this one [8]. Secondly, there is the area of broader partner management, which focuses on how an organization must organize its internal infrastructures to accommodate partners optimally, address their requests, and provide them with a consistent interface to a well oiled organization, as addressed in this article. Finally, we see partner management on the broader scale of the SECO, i.e., how to organize not one set of partners, but the full set of partners in an ecosystem, from both the technical and the business standpoints. This is discussed in previous work as well [39].

There are many qualitative single case studies in this domain [L12,4]. In these qualitative studies, researchers outline the different goals partners and orchestrators have. However, it remains unclear what the main orchestrator goals, ecosystem enablers, and ecosystem partner management activities are, and how to create a partner management strategy that contributes to the health of the software ecosystem. Through this article, we create a common ground for partner management research by accumulating findings from a literature review in a single framework entitled the Software Ecosystem Partner Management Framework (SECO-PMF) and test our framework and method in a set of five industry focused exploratory case studies. By creating this common framework and vocabulary we contribute to understanding how governance mechanisms affect the success and health of ecosystems, as pointed out in the ecosystem governance research agenda by Alves et al. [2], and provide aid in developing ecosystem orchestration methods, which aligns with the software ecosystems research agenda [43].

[1] https://newsroom.intel.com/editorials/intel-invests-1-billion-ai-ecosystem-fuel-adoption-product-innovation/#gs.tqi5ot

The SECO-PMF is evaluated through five exploratory case studies of orchestrators, in the bookkeeping and accounting software market of the Netherlands. We investigate orchestrators and ecosystems of varying size and maturity. By using this approach, the framework's completeness and applicability can be evaluated in scoped but varying contexts. Finally, based on the case study findings and academic expert evaluation, we adapt the framework. In this research, we provide the following contributions:

1. The SECO-PMF categorizes orchestrator activities, the creation of ecosystem enablers, and outlines methods for satisfaction of partner goals in section 3.
2. An evaluation of the SECO-PMF in a multiple case study of five SPOs and a partner management assessment in Section 4.
3. The partner management perspective that allows researchers to further contextualize partner management practices in Section 7.1.
4. A methodology for evaluating orchestrator's partner management practices in Section 7.2.

We elaborate on the framework adaptation in Section 6, reflect on our research approach in Section 8 and generalize the findings and suggest further research in the conclusion (Section 9).

## 2. Research Method

Design science [33] is applied to create a framework for evaluating orchestrator practices in partner management. We highlight the six steps of design science in Figure 1. We address the research question: ***How can orchestrators evaluate their partner management practices in software ecosystems?*** In understanding the components of partner management in software ecosystems, we first select relevant literature. Subsequently, we create a version of the SECO-PMF from the literature. Thirdly, we evaluate and improve the framework through five industry case studies of existing software ecosystem orchestrators.

| Design Science Process Step | Problem Identification | Derive Solution Objectives | Design and Development | Demonstration | Evaluation | Communication |
|---|---|---|---|---|---|---|
| | Orchestrators do not possess sufficient knowledge on how to evaluate their partner management activities in software ecosystems. | With a framework, organizations should be able to better evaluate partner management activities and software ecosystem orchestration. | Through a snowballing literature study, we identify the main components of a SECO-PMF. | The framework is applied in five case studies and continuously improved. | The case participants and academic experts evaluate the use of the SECO-PMF and propose improvements. | The framework is published and its use is promoted. |
| **Research Step** | Evaluate whether this is a problem within case studies. | Reason how a solution could contribute to better software ecosystem orchestration. | Snowballing literature study. | Application in five empirical case studies of orchestrators. | Evaluation with case study participants and academic experts: does the framework reach its objective? | Submit work to excellent journal and promote framework to industry contacts. |
| **Addressed in** | **Section 1** | **Section 1** | **Sections 2.1, 3** | **Section 2.2, 4** | **Section 2.3, 5, 6** | **Complete Article** |

Figure 1: The design science method is followed in this work as the umbrella method. In more detail, we also apply a snowballing literature survey and five exploratory case studies, in which we explore the artifact's effectiveness with the case participants.

### *2.1. Literature Mapping Process*

The SECO-PMF primarily consists of activities that orchestrators can perform. To seed this list of activities, a literature mapping was conducted according to the mapping guidelines of Petersen, Vakkalanka and Kuzniarz [50]. A literature mapping approach was selected to ensure the first set of activities would come from modern literature, has been described extensively, and would not be biased by the researchers' points of view or the organizations participating in the study. We report according to the systematic review guidelines for engineering research of the ACM guidelines [53].

In conducting the literature mapping, we used the Publish or Perish (PoP) software[2]. PoP enables large scale collection of meta-data of studies, which helps us in identifying articles in the research field. We use a tool because search engines such as Scholar and Scopus do not have an interface suitable for bibliometric analyses [31]. PoP is frequently used for extracting results from search engines, as there are currently nearly 1,000 published papers referring to the tool [31].

Although the tool enabled us to quickly collect the required studies, it does come with several drawbacks. Namely, the tool suffers from the selection biases of the search engines that it extracts data from and it can only be used with Scholar and Scopus (at the time of extraction). To overcome these limits, we conducted backward and forward snowballing on the collected studies.

Based on the terms management, partner and software ecosystem, terms defined in the introduction, we created search queries. Searching for studies required multiple synonyms due to the varied use of terms in the domain of software ecosystems. Therefore, we searched with the following query: (*management* OR *power* OR *value*) AND (*partner* OR *complementor* OR *alliance*) AND (*software ecosystem* OR *business* OR *ecosystem*).

By using PoP on Google Scholar and Scopus, we collected a literature set of 120 studies concerning ecosystems.[3] We limited the number of collected studies to put more emphasis on snowballing for data collection. After gathering the literature list, we formulated exclusion criteria to differentiate between studies that do not match the scope of this research and studies relevant for framework creation. We excluded studies that meet one of the following exclusion criteria:

1. Not concerning partner management in software ecosystems. This was selected as exclusion criterion because it is the scope of the research (removed 88);
2. Not investigating ecosystems around a software product. Non-software ecosystems are not designed to manage the specific characteristics that come with software product collaboration, such as modularity of architecture and standardization of products and interfaces [62,81] (removed 11);
3. Not reporting concrete management activities. This limits the relevance of a study for the purpose of framework creation (removed 9);
4. Only investigating young software ecosystems without mature partner management presence. Studies that meet this criterion are excluded to limit the scope of partner management activities to those that have been successfully applied in practice (removed 3).

The elimination led to a literature list of nine studies (8%). We provide the exclusion reason for every study in the data set listed above. To overcome the tool and search engine bias, we complemented the query search with backward and forward snowballing. This may identify a larger share of relevant literature than query search alone [77]. Because of the varied use of terms in software ecosystems, solely relying on queries may miss essential literature that named its concepts differently.

We have used the snowballing guidelines described by Wohlin [77]. The forward snowballing was conducted on the nine studies by using the "cited by" feature. Backward search was performed by investigating all citations in the bibliographies of the identified studies. This led to a literature set of thirteen articles, listed next to the references.

The final literature mapping process concerns the identification of key concepts in the selected studies. To mitigate researcher bias, we created a meta-model of the key concepts in partner management from established literature, seen in Figure 2. The model is based on the meta-model assembly technique described by Brinkkemper et al. [14]. The meta-model acts as a coding scheme to categorize the concepts, their relations and other findings in literature. It has acted as the foundation for creating the SECO-PMF. In short, the meta-model describes the orchestrator, its partners, and the partner management activities that the orchestrator can perform to enable partners to reach their goals. The contents of the meta-model are elaborated in Section 3.3.

[2]The tool is available at `https://harzing.com/resources/publish-or-perish`

[3]The literature list data is accessible online at Mendeley Data [69].

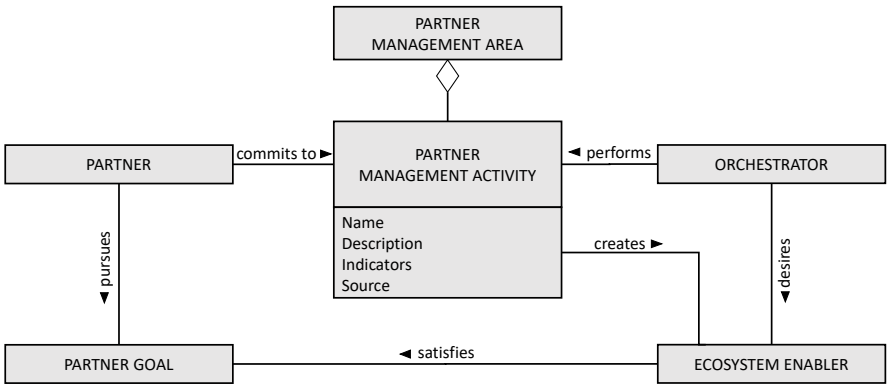



Figure 2: Meta-model that categorizes key concepts of partner management and their relations, based on the works of Rickmann et al. and Selander et al. The Meta-model is described in detail in Section 3.3.

| Study | Partner Goal | Ecosystem Enabler | Orchestrator Activity | Ecosystem in scope |
|---|---|---|---|---|
| [L1] | Dependent variables | - | Instrumental variables | SAP |
| [L2] | - | Power and trust | Complementor force | Enterprise application software |
| [L3] | - | Construct | Capability | Business applications software |
| [L4] | Result | Value Capture | Value creation | Open source software ecosystems |
| [L5] | Goal | Enabler | Instrument | SAP |
| [L6] | Complementor change | - | platform governance decisions | Kiva |
| [L7] | Business goal | - | Mechanism | Mobile technology and service |
| [L8] | - | Mechanism | Manifestation | Enterprise software solutions |
| [L9] | - | Value competition | Ecosystem relation government | Sony Ericsson |
| [L10] | - | Performance | Network governance mechanism | Open source software ecosystems |
| [L11] | - | Facilitator | Strategies to Foster Partnerships | ERP software |
| [L12] | - | Ecosystem Advantage | Keys | ARM |
| [L13] | - | Bridge | Cause | Axis |

Table 1: Literature list and identified concepts. The table shows the concepts that have been marked as partner goals, ecosystem enablers and activities. These will be used in the combination process.

Since studies use various terms when referring to a concept, the meta-model acts as a coding scheme for categorizing orchestrator activities, partner management enablers and partner goals. Assigning codes to concepts in the studies was performed in two ways. In 8 out of 13 articles, a model visualized the relationships between the concepts of the study. The model allowed us to code model elements as activities, ecosystem enablers and goals in the corresponding study. In articles without a model, the activities, enablers and goals were identified by coding based on sentence structure. Concepts were coded as activities when the concept is a verb and the orchestrator is the subject of the sentence. For example, when an article contains the text *The orchestrator does X*, we list X as a partner management activity. In a similar manner, we coded the ecosystem enablers and the partner goals. For every identified activity, enabler or partner goal, we also extracted its relationship with the other concepts.

Table 1 contains the overview of the concepts that were coded as activities, ecosystem enablers, and partner goals per study. For example, the work of Selander et al., [L9], uses the term "ecosystem relation government" for activities and ecosystem enablers are named "value competition."[4]

### *2.2. Partner Management Framework Creation*

After identifying existing concepts that would populate the SECO-PMF, the identified concepts were classified for standardization and framework creation. First, clusters were defined in the goals and ecosystem enablers list by grouping similar concepts. For example, "meet customer needs" and "customer demand" fall in the same cluster. After establishing the clusters, each cluster was named as a single goal or ecosystem

[4]The data from the process of concept identification is accessible online at Mendeley Data [68].

| SPO | Founding | employees | partners | connections | customers | Market Segment | Interviewee role |
|---|---|---|---|---|---|---|---|
| A | 1981 | 50-100 | 30 | 90 | 8.000 | SMEs | Partner manager |
| B | 1984 | 100-200 | 150 | 400 | 70 | Entrepreneurs | Partner manager |
| C | 1980 | 100-200* | 150 | N/A | N/A | SMEs | Partner manager |
| D | 1987 | 100-200* | 80 | 500 | 15.000 | Accountants and bookkeepers | Team leader partner management |
| E | 1984 | 1000-2000 | 500 | 500 | 400 | SMEs | Director ecosystem |

Table 2: Descriptives of the case study companies. (*) Company C and D are part of larger software conglomerates, the indicated number is the size of the bookkeeping and accounting software branch.

enabler. The identified activities were grouped into clusters. Activities related to a specific reason for the orchestrator and partner to interact were also identified. Cluster titles were used, such as marketing, innovation, or road–mapping. To structure the framework, the clusters were divided across the governance mechanisms of Alves et al. [2]. We merged similar activities and created a list of unique activities. If a study mentioned that a certain activity would influence an ecosystem enabler, that relationship was adopted by the SECO-PMF. Therefore, the SECO-PMF shows the links between activities and ecosystem enablers, and between ecosystem enablers and partner goals.

A total of 21 goals,[5] 37 enablers,[6] and 107 activities[7] have been identified in the literature set. The activities were clustered, merged, and removed in case of redundancy.

For activities, fifteen clusters were created as part of the clustering process. Equivalent activities were merged iteratively. This reduced the number of activities to 48. After establishing activities, ecosystem enablers were aggregated and extracted to reveal what effect an activity has. In this way, the effect of an activity can be assessed towards partner goals. The opposite direction is also possible. When an orchestrator is willing to satisfy a partner goal, it can see which ecosystem enablers satisfy this goal and which activities provide these ecosystem enablers.

### *2.3. Framework Evaluation by Case Study Investigation*

After framework design and population, we performed a framework evaluation at five SPOs in the Netherlands. Following the case study methodology of Yin [80], we performed a multiple case study, embedded in the context of the Dutch bookkeeping and accounting market. This market was relevant for this research because it consists of multiple software ecosystems where orchestrators compete for customers and partners. Some partners offer their added value in multiple ecosystems, where others are bound to a single orchestrator and ecosystem. We define the bookkeeping and accounting market as multiple ecosystems, as the actors do not share one platform.

By performing the investigations in a single market, we conceived understanding of partner management in a single domain from multiple perspectives, similar to the approach of Scholz [57], leading to a more in depth perspective of partner management. While we could have decided to study platforms from multiple domains, we hypothesized that studying one relatively mature software market would lead to more detailed partner management activities.

Several suitable markets were identified in which products have become so commoditized that competition is for a large part determined by the strength of the ecosystem; customer relationship management, ticketing software, and enterprise resource planning software. For each of these markets, however, there were few companies that were both available and accessible for our research. The one market that has these characteristics and is also highly localized is the bookkeeping and accounting market, in large part because accounting is a domain that historically grew in the local fiscal context. The market fit our criteria: highly commoditized, many local players, and multiple large and healthy ecosystems.

In the bookkeeping and accounting market, the ten largest players were invited to take part in this research. The ten largest players were selected from a market report [13]. Five organizations responded positively and were selected. The case study approach allowed thorough investigation of the orchestrator's

[5]The list of identified goals is accessible online in the "Clustering of goals" Section at Mendeley Data [68].
[6]See section "All Enablers", ibid.
[7]See Section "All Activities", ibid.

partner management, where the framework was used to frame the activities of the case companies. Yin's guidelines in the design, execution, and reporting of the case studies [80] were followed.

The case protocol consisted of a document study part and an extensive interview part. We conducted document studies to gain understanding of the ecosystems within the broad social context, perform cross-validation with the interviews, and complement the interview findings. The document study was performed by studying corporate websites, news websites, partnership agreements, and app stores of the case companies. The process of document study was performed as a preparation to the interviews by the researchers. In some cases, we referred to the documentation during the interviews, but the document study had as its main aim to understand the background of the organization, the platform, and the manner in which the orchestrator organized its partner management beforehand.

Secondly, the interview protocol was formulated to investigate the partner management strategies of the orchestrators. The protocol consisted of a general investigation of the partner management strategy, and an inquiry in the specific partner management activities. For the first part of the interview, we created a generic list of interview questions, supplemented with results of the document study that require further validation. For the second interview part, we adapted the activities to an interview protocol by forming a question about each activity. For example, an activity called "create new partnerships" can be investigated by asking: "How do you attract new partners?" Finally, we investigated activities that are not present in the SECO-PMF, but that the company may use, as this question could reveal missing elements in the model.

A semi-structured interview approach was used based on an interview protocol, with space for follow-up questions.[8] The semi-structured interview protocol allowed for coding per question. All answers to a question fall under the same code. After codifying the transcriptions and combining document study results with interview results, we could analyze the case study findings.[9]

During the interviews we discussed the practical application of every partner management area; in this article, quotes are provided to clarify the orchestrators' positions. The interviews lasted for 92 minutes on average, in which the PMAs were extensively discussed. The interviews were recorded and transcribed manually within 24 hours after the interview. After the interviews, the framework was adapted based on the interviewees comments.

### *2.4. Framework Finalization through Expert Evaluation*

The final version of the SECO-PMF is created by adapting the framework based on two sets of interviews. The first set of interviews was done with the case study interviewees to establish whether they all accepted the last version of the PMF or whether there were still comments they wished to see included. Furthermore, we invited four academic experts on software ecosystems to perform an expert review of the framework as a finalization step. The interview protocol and process are described in Section 6.

Both the academic experts and the case study interviewees approved of the results and confirmed the changes to the model that were introduced during the five case studies. Furthermore, we find that new work coming out on the topic confirms to a large extent the categories of PMAs that we find in this work [9].

The framework adaptations proposed during the interviews and implemented in the final version of the SECO-PMF are listed in Section 6.

## 3. Framework Background, Rationale, and Content

This section serves three purposes. First, we discuss previous literature regarding the actors in ecosystems and the relevance of partner management. Secondly, we describe how the meta-model and the SECO-PMF classify the concepts and relations in this context. Finally, we discuss the contents of the SECO-PMF as well as its partner goals, ecosystem enablers, partner management areas and activities.

[8]The interview protocol is available at: `https://bit.ly/2EF4veK`

[9]The interview transcripts are available from the authors upon request.

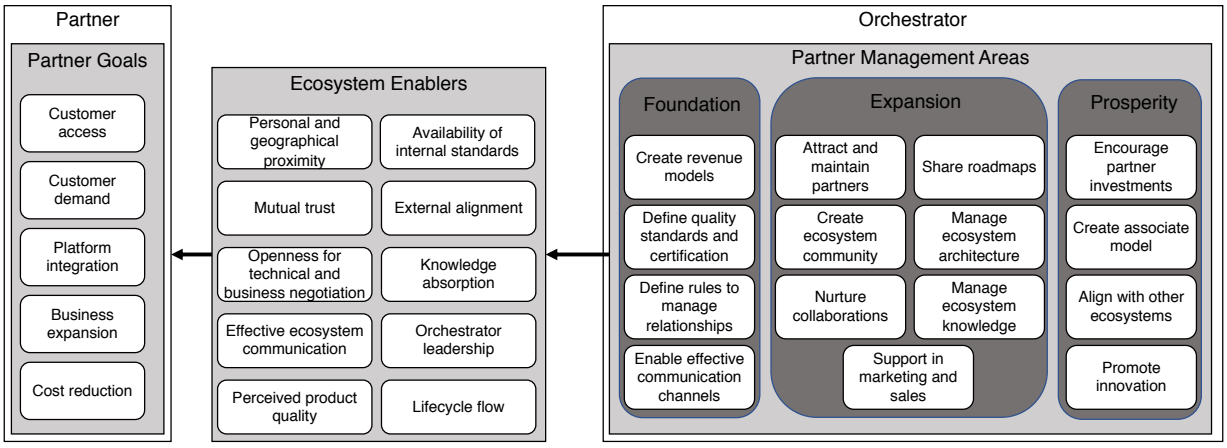



Figure 3: The Partner Management Framework (SECO-PMF) for orchestrators that shows the partner goals, ecosystem enablers and partner management activities. For readability concerns, the partner management activities have been omitted from this view.

### *3.1. Actors in Software Ecosystems*

Within the software ecosystems under study, three types of actors are identified: ORCHESTRATORS, COMPLEMENTORS, and CUSTOMERS [35]. A complementor or niche player provides a specialized product or service that complements the platform provided by the orchestrator [15,37]. For example, Rovio entertainment provides mobile games such as Angry Birds, to extend the product offering of the ecosystem it participates in [79]. By doing so, Apple can abstain from investing R&D resources in developing mobile entertainment. Therefore, diverting part of the platform's value proposition to complementors can attract new customers interested in the platform extension, while also satisfying existing customers' needs by adding new functionalities [L12].

Different roles exist in the group of partnerships that an orchestrator forms on its platform [30]. These are:

1. **Software vendors** develop value proposition extensions, such as an app developer for mobile ecosystems;
2. **Service providers** facilitate access to customers for the software ecosystem but do not develop extensions themselves. Examples are implementation, configuration, reselling, or consulting companies;
3. **Infrastructure providers** do not sell to customers but provide services essential to the platform. For example, software ecosystems that use a cloud-based delivery model for delivering software to customers may be dependent on Microsoft for providing the Azure platform.

Not all complementors in an ecosystem should be seen as partners. Complementors may be unknown, or they are known and the orchestrator is not maintaining a relationship with them. In this research, we focus on the complementors that have explicit relationships, for instance through contracts, with the orchestrator. Complementors that do not have any formal relationship to the orchestrator were considered out of scope for this research.

### *3.2. Relevance of Partner Management*

SOFTWARE ECOSYSTEM GOVERNANCE consists of the activities by which an orchestrator creates value, coordinates relationships, and defines control [2]. One could reason that it was poor software ecosystem governance that led to the demise of Symbian [76], as the team behind Symbian in rapid succession lost support from major partners, such as Samsung and Sony–Ericsson, and did not manage to generate new

support for its then leading mobile operating system. Software ecosystem governance is essential for the existence of a healthy and sustainable ecosystem. For the orchestrating company, governance mechanisms are life-or-death decisions, as a failing ecosystem can also bring down its leader [35]. There is a need for practical governance guidance to aid companies in ecosystem governance [2].

We consider partner management and ecosystem governance as closely related subjects. However, there is a difference. Partner management is more narrowly focused on establishing relationships with partners and to allow thriving partners to improve the ecosystem [L12]. Ecosystem governance concerns the comprehensive subject of ecosystem health, which is also affected by customers, technologies, and establishing power between actors [42]. To further emphasize the difference, partner management is part of ecosystem governance, but has relationships with partners as a focal point, instead of ecosystem health.

The orchestrator can meet customer demand for complex, integrated solutions by utilizing partners' capabilities while maintaining a focused platform [28,61]. By using partners' domain knowledge, markets may be penetrated that would not be accessible without partner support. Therefore, the use of partners enables a platform to provide value for a wider variety of customers, leading to more revenue for orchestrators. For example, SAP has partnerships that extend the proposition of the SAP ecosystem[66], such as Red Hat for Linux and Kubernetes technology. This enables the SAP ecosystem to serve customers that demand these technologies while SAP can limit itself on developing the platform.

When dividing responsibility in an ecosystem, the orchestrator can construct economies of scale, as every developing company in the ecosystem is able to specialize in its complement. When an actor specializes in a part of the value proposition, economies of scale lead to a cost reduction [28]. This is beneficial for the actors in the ecosystem. Therefore, the orchestrator can cut development and maintenance costs when solely focusing on the platform. For example, banks experience a cost reduction and a performance improvement when outsourcing part of their IT infrastructure [59].

Gawer states that using partners to extend the platform enables the use of partner resources [24]. Partners possess strategic investments, innovations, specialist capabilities, and intellectual property that an orchestrator can use [L12]. These resources can help the orchestrator overcome entry barriers of new markets. Thus, the orchestrator can grow its customer base in new markets by using partners.

However, there are reasons for abstaining from partner attraction. A software company that refrains from using partners can stay in managerial control. The use of partners may lead to threats of security and quality issues [59]. Additionally, evolving into an industry platform from an internally developed product increases the number of actors in the ecosystem who will bring both innovative capabilities and competitive capabilities, as stated by Gawer. These actors may start innovating in ways that become competitive to the platform. The emergence of competition from complementors depends on ecosystem governance. Failing to effectively govern the ecosystem may result in intense competition and a downfall of the ecosystem [2]. An interesting example is found in Windows Mobile. When Microsoft first launched its mobile OS, they immediately approached Facebook to create a Facebook App for that platform. Facebook refused, stating that Microsoft could easily build the Facebook app themselves, using Facebook APIs, which Microsoft eventually did. However, Gawer concludes that effective partner management can mitigate competition between partners and increase complementors' incentives to innovate in platform-enhancing ways.

### *3.3. Meta-model for Partner Management*

Based on the literature identification process, we introduce a meta-model for this research, shown in Figure 2. The meta-model illustrates the relevant concepts in this research, together with the relationships among them. It is aligned with the identified research gap. Both Rickmann et al. [L5] and Selander et al. [L9] have created a similar model, and we adopt their work for this research. Both studies model the relationship between the orchestrator and its partner in three components. Namely, it is modeled in (1) an orchestrator activity, (2) a result of the activity, and (3) how the partner benefits from the result. This abstraction is also used in the SECO-PMF.

The meta-model describes the collaboration between ORCHESTRATOR and PARTNERS. A PARTNER MANAGEMENT ACTIVITY, activity for short, is an interaction between orchestrator and partner [L9]. Within the scope of this research, the orchestrator carries out activities, activities create ecosystem enablers and enablers satisfy partner goals.

We cluster the activities in PARTNER MANAGEMENT AREAS (PMAs), as PMAs contain activities that focus on the same business area. The ORCHESTRATOR is the founding actor in the ecosystem [21,20,55]. The orchestrator selects activities to unlock the potential of partners and the ecosystem [19]. Different activities lead to different enablers to committing partners, which leads to the satisfaction of other partner goals [L5]. Therefore, when an orchestrator selects the right activities for its goal, it influences how partners engage and how the ecosystem grows [L8].

An ECOSYSTEM ENABLER is a benefit to a business goal that is derived from other actors[L5]. Ecosystem enablers include economies of scale, knowledge pooling, and increased learning [L13]. Partners create these enablers by being active in the ecosystem, such as by providing knowledge, selling to customers, or assisting in innovation [28]. Orchestrators desire ecosystem enablers, as they can be seen as building blocks for establishing sustainable relations with partners [65]. Knowledge and trust are ecosystem enablers [L2], as they come from the interaction with other actors in the ecosystem and support a partner in the achievement of its goals [5]. When an orchestrator uses activities that hinder the satisfaction of partner goals, it constructs entry barriers, that discourage partners to join the ecosystem [42]. An example of a barrier is a participation fee, which is beneficial for the orchestrator while hindering partners.

A PARTNER GOAL is an objective that a partner sets and achieves by participating in the software ecosystem [52]. The partner achieves goals by profiting from ECOSYSTEM ENABLERS [2]. The meta-model can be interpreted as follows. A PARTNER extends the ecosystem's value proposition and has a bidirectional relationship with the ORCHESTRATOR. It maintains the relationship with the ORCHESTRATOR and the ecosystem as these provide ECOSYSTEM ENABLERS that the PARTNER benefits from. If a partner decides to commit to an activity, it benefits from the enablers, which satisfy its partner goals. The orchestrator is therefore encouraged to perform activities, as they ultimately attract partners. These activities, ecosystem enablers, and partner goals together with the relations among them is what we named SECO-PMF. Its creation and evaluation is the main goal of this research. The concepts in the tables and figures in this article hereafter are consistent with this meta-model.

### *3.4. Positioning the SECO-PMF among Existing Literature*

The meta-model that acts as the foundation of this research has been filled with the literature results and model creation results in Figure 3. The model illustrates the partner, orchestrator, and the ecosystem boundary. For clarity concerns, one partner is shown in this figure. We start the SECO-PMF's description with the goals in the next section. Afterwards, we describe the ecosystem enablers, PMAs and activities.

This research is rooted in several foundational theories of information systems (IS). First, Actor Network Theory [16] stimulates us to study the heterogeneous actors that operate in organization, such as a software ecosystem, and in this article we study partners, orchestrators, and customers alike. Secondly, we value the whole network as per Social Network Theory [36], and look at both the interconnections between the orchestrator and partners and between partners themselves. Thirdly, we value Boundary Resource Theory [22], as the interfaces that are made available by the orchestrator to partners enable them to collaboratively achieve network goals.

This article contributes to IS theories in the following ways. First, this article reconfirms the importance of studying multiple types of actors in software ecosystems. For Boundary Resource Theory we focus on the role of the app stores and establish that app stores are perhaps one of the most important boundary resources available, if not belonging to a whole new concept of boundary resource marketplace. Finally, we extend the SECO meta-model of Wouters [78] with partner management activities, with the goal of emphasizing the role of partner management (activities) in software ecosystems.

Partner management can be investigated at an abstract level. Williamson and De Meyer perform a thorough case study investigation and provide six keys to ecosystem advantage [L12]. These abstract high-level partner management keys are relevant and important in ecosystems. We have adopted these keys and attempted to provide concrete practices.

The partner management model described by Rickmann et al. consists of goals, ecosystem enablers, and instruments [L5]. The model describes five instruments that were identified in a case study. The article puts forward that ecosystem governance should be investigated from the perspective of the partner. The model

describes the partner lifecycle and investigates a single relation between the orchestrator and a partner. In this research, we have adopted the perspective of the partner by exploring ecosystem enablers and partner goals. We investigate the effectiveness of partner management not as individual relations, but in the ability of the SPO to orchestrate the entire ecosystem. We hypothesize that an ecosystem does not rise or fall with individual partner relations, but with the combined effects of the relations between ecosystem actors.

We contrast our work to several existing works that also explore the creation and theoretical analysis of partnership models in Table 3. We first identify their *focus* and try to delineate their scope. Secondly, we identify the main *categories* that are being used to identify the main mechanisms and attributes needed to design the model. Thirdly, we identify and list the *attributes* that make up the model, which are typically extensive in number. We *contrast* these competing models against our own and identify why and when our model is more appropriate to use. In the column *case studies* we list the number of case studies used to evaluate the models in the different studies, i.e., at least one per study, illustrating the empirical nature of these works. Subsequently, we define which *concepts* are covered by the different models and show that our work is the more "complete" of the five. Finally, we identify the concepts that are not present in our model but are mentioned by other models.

We find that there is a clear path of creation from the more abstract models of Rickmann et al. [L5], to the more practical models of Jansen [39] and Belo and Alves [9]. We also observe that the models of Belo and Alves include a more strategic view: a topic that in our model is also underemphasized and admittedly deserves more attention, as an orchestrator's strategy strongly influences the composition of the partners that are accepted into a partnership program. We also find that these models wrestle with two things: first the role of customers is underemphasized, although every partner readily admits that customer access is the main driver for their participation in a partner program. Furthermore, these models include technical aspects, but cannot go too deeply into the topic. The building of theoretical bridges between the technical aspects of partner management and their business impacts will remain an academic research challenge for the next decade.

### *3.5. Partner Goals and Enablers in Software Ecosystems*

To attract partners, the orchestrator should understand why partners join a software ecosystem, and which benefits they expect to receive. The ecosystem should support the partner in meeting business goals. If the orchestrator and the ecosystem fail in supporting these goals, then the partner has no reason to stay and will leave the ecosystem [L5]. On the other hand, if the ecosystem supports a partner in reaching its business goals, that partner has all the reasons to maintain its relationship with the orchestrator. Rickmann et al. distinguish four partner goals, namely, Access to customers, Customer demand, Platform integration, and Business expansion. The work of Morgan et al. adds another distinct partner goal, Cost reduction [L4].

The business goals of partners can align strongly with orchestrator goals. For instance, a partner may advise customers to start using a particular platform so that customers can use the platform and the partner's extension. In this case, the orchestrator acquires another customer for the ecosystem. Partners have varying strategies for growing a business and making a profit, and therefore set varying goals when participating in a software ecosystem.

The partner goals are shown on the left in Figure 3 and in Table 4, along with a description. Therefore, we assembled a list of five partner goals that appear in software ecosystems. We added the research articles in which we identified this goal and provided the list of ecosystem enablers that satisfy this goal. All partners goals that have been found in the literature set fall in one of the five categories.

Distinguishing partner goals allows for an investigation into the strategy of partners and it allows these companies to identify which aspects of doing business may be limiting their profitability. An orchestrator that understands why partners maintain their relationship can adapt its partner management accordingly.

To reach these goals, the orchestrator can help partners conduct activities that positively contribute to ecosystem enablers. Varying strategies and ecosystems require different ecosystem enablers. Valença, Alves and Jansen investigated the facilitators of managing partner relationships [L11]. These facilitators contribute to the formation and growth of partnerships. This is closely related to the concept of ecosystem enablers. Hence, we take the facilitators as a basis for categorizing the ecosystem enablers. Valença et al.

| Ref | Title | Focus | Categories | Attributes | Contrast to this work | Case Studies | Coverage of model concepts | | | | | | Other concepts |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | P. Mgmt. Area | P. Mgmt. Act. | Partner | Orchestrator | P. Goal | Eco. Enabler | |
| **This work** | The Orchestrator's Partner Management Framework for Software Ecosystem | Partnership management, introduction of partner goals and orchestrator goals alignment | Foundation, Expansion, Prosperity | Create revenue models, define quality and standards certification, define rules to manage relationships, enable effective communication channels; Attract and maintain partners, share roadmaps, create ecosystem community, manage ecosystem architecture, nurture collaborations, manage ecosystem knowledge, support marketing and sales; Encourage partner investments, create associate model, align with other ecosystems, promote innovation | None | 5 | ● | ● | ● | ● | ● | ● | None |
| **[39]** | S. Jansen. A focus area maturity model for software ecosystem governance. Information and Software Technology, 118:106–219, 2020. | FAMM, hybrid between technical and management concepts | Ecosystem health, Open Markets, Intellectual Property, Open Platforms, Open Innovation, Software development governance, Associate Modesl | Partner grooming, Partnerships, Consulting partner support, Connect customers and partners, Marketing and sales, Training, Sales partner support; App testing, Application quality, Developer relationships, Process automation, Development partner support, Requirements sharing, Roadmapping, Developer monitoring; App market, Application format and delivery, App approval process, App curation, App marketing, Community Engagement, Business models; Licensing, Digital asset management, Patent management; Platform hardening, Platform extensibility, Software operation knowledge, Platform documentation, Security, Platform evolution; Competing ecosystem analysis, Market and customer analysis, Partner health assessment; Standards participation, Partnering with academia, Inspiration for developers, Open technology road maps | (+) Maturity model; (+) Easy to adopt; (+) Extensive empirical evidence; (-) Serves both technical and non-technical audience; | 12 | ● | ● | ○ | ● | ⊙ | ○ | Management maturity, Technical dimensions |
| **[9]** | Í. Belo and C. Alves. How to create a software ecosystem? a partnership meta-model and strategic patterns. Information, 12(6):240, 2021. | Providing a theoretical model in which to place strategic patterns. Meta-model for partner management analysis. | Platform management, Document management, Risk Management, Conflict management, Partner cluster management | Partner training, partner performance, marketing support strategy, communication strategy, sales channels and distribution, monetization strategy, network effects strategy, partners profile definition, keystone performance, platform evolution strategy, platform integration strategy, product and service homologation, document, document management strategy, risk, risk treatment, conflict, conflict treatment | (+) Grounded theory in empirical evidence; (+) Combines theory and practical evidence; (-) Includes very concrete aspects in combination with highly theoretical aspects; (-) Hard to adopt for practitioners | 3 | ● | ● | ● | ● | ● | ⊙ | Strategy, Conflict management, Risk management, Technical dimensions |
| **[L5]** | T. Rickmann, S. Wenzel, and K. Fischbach. Software ecosystem orchestration: the perspective of complementors. Twentieth Americas Conference on Information Systems (AMCIS), 2014. | A first attempt at a theoretical model of partner management, using grounded theory. | Goals, Enablers, Instruments, Effects, Intentions, Influencers | Customer access, Customer demand, Integration, Expansion; Resource sharing, Leadership, Lifecycle flow; Information resources, Personal contact, Infrastructure tools, Training, Formal agreement; Perceived usefulness, Satisfaction; Intention to partner, Intention to continue partnership; Ecosystem experience, Trust | (+) Grounded theory in empirical evidence; (+) Foundation for this work; (-) Relatively abstract; (-) Impossible to use for organizations; | 1 | ○ | ⊙ | ● | ● | ● | ● | Intentions, Influencers, Customer perspective |
| **[L13]** | K. Wnuk, P. Runeson, M. Lantz, and O. Weijden. Bridges and barriers to hardware-dependent software ecosystem participation–a case study. Information and Software Technology, 56(11):1493–1507, 2014. | An empirical study of partnering and identifying a set of bridges and barriers. | Bridges (reasons), Bridges (benefits), Barriers (hesitations), Barriers (drawbacks) | End customer demand, Relationship, Open environment, Geography, External standard, Future possibilities, Marketing, Low risk; Internal standard, Ease of installation, External standard, Less infrastructure, Scalability, Fewer resources; Performance, Technical features, Business model, Unclear roles, Performance, Lack of information about customers, Maintenance; Performance, Debugging, Information-gap, Compatibility; | (+) Hardware also considered, (+) Empirical evidence, (-) Poor theoretical analysis, (-) Limited in theoretical and empirical scope | 1 | ○ | ○ | ⊙ | ⊙ | ⊙ | ● | Bridges, Barriers |

Table 3: A comparison of our work is shown to existing works, and how well those works cover the meta-model presented in this work. It is interesting to note that two articles explicitly address a technical dimension as well. Furthermore, the customer perspective appears to be underemphasized in most works, including this one, even though customer access is one of the main motivators for partners to participate in a partner program. ○ stands for "no coverage", ⊙ stands for "some coverage", and ● stands for "complete covers the concept".

| | Partner Goal | Description | Source | Is satisfied by ecosystem enabler |
|---|---|---|---|---|
| PG1 | Customer access | Partners that have the goal of achieving customer access want to increase the visibility to customers, as this extends the partners' customer base. The partner also requires the sales channels that provide access to customers. | [L4,5] | E2, E3, E5, E6*, E9 |
| PG2 | Customer demand | The current customers of an ecosystem may have unsatisfied needs. Partners that have the goal of meeting customer demand want to increase sales by joining the ecosystem to meet this need. This involves gaining knowledge about customer demands and being able to adapt the products based on these demands. | [L1,4,5,9, 7] | E3, E4*, E6, E7, E8, E9, E10 |
| PG3 | Platform integration | A software company that wants to profit from resources in the ecosystem should connect to the orchestrator. Integration of a software product with the core platform is a reason to participate in a software ecosystem. | [L5] | E1*, E2, E4*, E6, E10 |
| PG4 | Business expansion | Business expansion for partners is growth of the company by ecosystem resources. Growth manifests itself in the number of developed products, the number of innovations, the amount of content, the amount of profits generated, and the market value of the company. | [L6,5,9] | E1*, E2, E5, E7*, E8, E10 |
| PG5 | Cost reduction | A partner may further increase performance by reducing the costs of producing and maintaining software. By participating in a software ecosystem, a partner may benefit from economies of scale and other effects that reduce the costs that the company experiences. | [L4] | E3, E7*, E9 |

Table 4: The five partner goals in software ecosystems. (*) Indicates that the enabler-activity relationships were not found in theory but were observed in the case studies.

distinguish seven facilitators of which we adopt six in the SECO-PMF, as "Respectful attitude" was not found in the literature set.

In addition to these six ecosystem enablers retrieved from Valença, Alves and Jansen, we identify four more ecosystem enablers in the literature set. Ecosystem enablers 7 to 10 do not directly foster the relationship between the partner and the orchestrator, but nonetheless provide value for partners. That might explain why [L11] did not mention these. The ecosystem enablers stand in the middle of Figure 3. Table 5 describes the ecosystem enablers, along with the satisfied goals and the activities that lead to the ecosystem enablers.

### *3.6. Partner Management Areas and Activities*

The final section of the SECO-PMF are the activities in the PMAs. In order to categorize the activities, we used the governance mechanisms list from Alves, Oliviera, and Jansen [2]. We adapted this list of governance mechanisms to PMAs.

We have taken the list of governance mechanisms and removed manage licenses, manage resources, manage risks, manage expectations, and distribute power. These governance mechanisms appear to be irrelevant for partner management because they either do not positively influence enablers for partners or could not be aligned with concrete activities. Furthermore, we combined the governance mechanisms of *Create partnership models* and *Establish roles and responsibilities* as partners take a role and responsibility that aligns with the partnership. Taking responsibility in an ecosystem is an advanced form of partnership. We called the new PMA *Create associate model*. We also combine *Define entry requirements* and *Stimulate partner investments and share costs* into *Encourage partner investments*, as these mechanisms require actors to invest to participate in the ecosystem.

Finally, we found activities that cannot be placed into the existing governance mechanisms. Therefore, we formed three new PMAs. *Align with other ecosystems* contains activities that generate value to partners by linking to other ecosystems. *Support in marketing and sales* contains activities that enable an orchestrator to aid partners in marketing and sales. Finally, *Create ecosystem community* concerns activities that turn an ecosystem from independent islands into a tightly knitted community. This left us with a list of 15 PMAs. These PMAs can be found in the SECO-PMF in Figure 3 and in Table 6.

We added an indicatory PMA categorization based on partner management maturity. The foundation column contains PMAs that establish the structure that facilitates partnerships. PMAs in the expansion column help the orchestrator in attracting partners and growing an ecosystem. The prosperity column

| | Ecosystem Enablers | Description | Source | Satisfies | Is provided by |
|---|---|---|---|---|---|
| E1 | Personal and geographical proximity | Physical proximity promotes joint projects among software companies. Companies that operate in the same region understand the specific needs of this market. Hence, geographically and personally close relationships between partners and orchestrators lead to easier integration and collaboration. | [L10,11] | PG3*, PG4* | A3, A7*, A12*, A13, A14* |
| E2 | Mutual trust | Trust results from the closeness of relationships between the orchestrator and the partner. It is seen as a premise for a partnership to emerge. Achieving a trustful relationship allows both companies to build on each other, and subsequently, reach goals where they need each other for. Furthermore, partners that enjoy trust from the orchestrator may be provided access to additional customer channels | [L13,5, 2,11] | PG1, PG3, PG4 | A1, A5, A6*, A8, A10, A12, A15, A18, A19, A20, A22*, A24*, A27*, A29, A31, A36, A47 |
| E3 | Openness for technical and business negotiation | Flexibility for business and technical negotiations is a critical factor for satisfaction of partner goals, as this enables the negotiation of a win-win approach. Therefore, setting an open environment for technical and business negotiation allows for communication, business opportunities, and cost reduction due to collaboration. | [L4,3,8, 11] | PG1, PG2, PG5 | A1, A7, A8, A9, A10, A12, A13*, A14*, A16, A17, A22, A25, A27, A28*, A29, A31, A33, A35, A37*, A38, A46, A47, A48 |
| E4 | Effective ecosystem communication | Communication is essential for a collaboration to succeed and thereby for a partner to derive value and satisfy its goals. Establishing adequate communication channels and ensuring that support for technical or business discussions is available enables a partner to integrate in the ecosystem and meet customer demand. | [L12,11] | PG2*, PG3* | A1, A2*, A5, A7*, A8, A10*, A13*, A14*, A17*, A28*, A29, A40 |
| E5 | Perceived product quality | The quality of products and services as perceived by customers is a criterion considered by orchestrators and partners to select new partnerships and for customers to adopt new products. Creating this ecosystem enabler eases the satisfaction of Customer access and generates opportunities for expansion. | [L13,5, 3,11] | PG1, PG4 | A3, A4*, A14*, A18*, A19, A20, A21, A22, A23*, A24, A25*, A26, A27, A42, A43, A43 |
| E6 | Availability of internal standards | By defining and forcing adoption of standards, the orchestrator facilitates the collaboration with partners. This eases the satisfaction of goals for which collaboration is essential. Partners connect easier with customers after aligning with the internal standards, and collaboration with other platform extensions is easier. | [L13,9, 11] | PG1*, PG2, PG3* | A1, A5, A10, A25*, A28*, A29, A31, A32*, A38, A39*, A45 |
| E7 | External alignment | The orchestrator has also set alignment with the upstream and downstream ecosystems. This eases the collaboration with actors outside the ecosystem. This helps in meeting customer demands but also expanding business processes upstream or downstream. | [L13,9] | PG2, PG4 | A31, A38, A43, A44 |
| E8 | Knowledge absorption | The orchestrator can aid in increasing learning of knowledge and can increase the availability of knowledge to partners. Doing so, partners may absorb more knowledge. This benefits partners in meeting customer demand and expanding their businesses. | [L12,4, 9] | PG2, PG4 | A2, A10, A11*, A12*, A13, A14, A15*, A16, A19*, A35, A36*, A37, A38, A42, A45*, A46*, A47, A48 |
| E9 | Orchestrator leadership | By creating a dominant vision in the ecosystem, the orchestrator can combine the individual powers of partners. This also benefits partners in reaching their goals. A strong leadership expressed by the orchestrator increases the committance of partners. Being led by a strong orchestrator vision enables partners to access customers, meet customer demand, and reduce costs by having a tightly bound ecosystem. | [L12,4, 5] | PG1, PG2, PG5 | A3*, A5, A6, A4*, A12, A17, A18*, A24*, A25, A26*, A29*, A33*, A34, A35, A37, A39, A40, A41, A45 |
| E10 | Lifecycle | Partners are not a homogeneous mass but are in different stages of the partner lifecycle. Recognizing the different needs of partners helps orchestrators to perform activities more specific to the partners lifecycle phase. Adapting activities aids partners in every lifecycle phase in reaching their goals as the orchestrator creates specific enablers for this partner. This aids partners in expanding their business. | [L5] | PG2, PG3, PG4 | A11*, A15*, A19, A20*, A23*, A27*, A28, A32, A34*, A36*, A39, A42*, A45 |

Table 5: Ecosystem enablers in software ecosystems. (*) Indicates that the enabler-activity relationships were not found in theory but observed in the case studies.

consists of PMAs that aid in maximizing the strength of existing relationships in the ecosystem, and ensuring long-term sustainability.

The entire list of activities is shown in Table 6. An activity can create multiple ecosystem enablers, as "the creation of ecosystem principles" leads to "trust", "effective ecosystem communication", "availability of internal standards" and "orchestrator leadership." The PMA with the largest number of activities is "support in marketing and sales" while several PMAs have only one activity. An orchestrator has numerous ways to assist partners in marketing and sales, while there are, for instance, few opportunities to manage conflicts.

*3.7. Software Ecosystems Scope*

In early work [43], different scope levels are defined: from simple software supply network perspectives that take into account only the suppliers and customers of an SPO, to the international scope of the "complete" software ecosystem, where all worldwide SPOs and related actors are connected in one big software producing and consuming hive. This article and the ones to which we compare our work (see Table 3) mostly focus on the supply network level.

This work is also scoped on the commercial software market. While not a dichotomy [58], open source is not considered in this work. We also must observe other types of SECOs, such as the mobile SECOs. These worldwide SECOs that address millions of mobile users rapidly outgrow the SECO-PMF that is presented in this work. Issues that for instance play a big part in our work, such as geographical proximity of the partner to the orchestrator, play a much smaller part in these worldwide mobile SECOs. While the orchestrators of

| | Partner management area | | Partner Management Activity | Sources | Ecosystem Enablers |
|---|---|---|---|---|---|
| Foundation | **Create revenue models** | A1 | Create revenue model | [L12,11] | E3, E6 |
| | | A2 | Enforce revenue streams | [L12,11] | E2, E4, E8 |
| | **Define quality standards and certifications** | A3 | Create partner certificates | [L10,6] | E1, E5, E9* |
| | | A4 | Certify partner applications | [L8] | E5*, E9* |
| | **Define rules to manage relationships** | A5 | Create ecosystem principles | [L2,7] | E2, E4, E6, E9 |
| | | A6 | Avoid orchestrator incursion by contracts | [L12] | E2*, E9 |
| | **Enable communication channels** | A7 | Set up personal partner contact | [L5,2,3,11] | E1*, E2, E3, E4* |
| | | A8 | Develop ecosystem communication channels | [L11,7] | E2, E3, E4 |
| | **Attract and maintain partners** | A9 | Create new partnerships | [L4,7] | E3, E10 |
| | | A10 | Discuss ecosystem value with partners | [L12,4,13,7] | E2, E3, E4*, E8 |
| | | A11 | Support non-technological partners in ecosystem | [L10] | E8*, E9, E10* |
| Expansion | **Create ecosystem community** | A12 | Create a community forum | [L5,4] | E1*, E2, E3, E8* |
| | | A13 | Create partner hubs | [L12,10] | E1, E3*, E4*, E8 |
| | | A14 | Organize conferences | [L4] | E1*, E2*, E4*, E5*, E8 |
| | **Nurture collaborations** | A15 | Train partners | [L5] | E2, E8*, E10* |
| | | A16 | Create shared R&D teams | [L10,6] | E3, E8 |
| | | A17 | Solve conflicts between partners | [L4] | E3, E4*, E9 |
| | | A18 | Point out excelling partners | [L5] | E2, E5*, E9* |
| | **Support in marketing and sales** | A19 | Share sales knowledge and channels | [L3] | E5, E10* |
| | | A20 | Share marketing channels | [L5] | E2, E5, E10* |
| | | A21 | Open platform to diverse markets | [L3,7] | E5 |
| | | A22 | Create joint marketing agreements | [L7] | E2*, E3, E5 |
| | | A23 | Create an app store | [L5] | E5*, E10 |
| | | A24 | Recommend partners to customers | [L3] | E2*, E5, E9* |
| | | A25 | Bundle third-party applications and platform | [L8,3,4] | E3, E5*, E6*, E9 |
| | | A26 | Create public recognition for the ecosystem | [L13] | E5, E9* |
| | | A27 | Share reputation with partners | [L3,8] | E2*, E3, E5, E10* |
| | **Share roadmaps** | A28 | Create ecosystem roadmap | [L12,4,7] | E3*, E4*, E6*, E9 |
| | | A29 | Force partners to align with roadmap | [L11] | E2, E3, E4, E6, E9* |
| | | A30 | Discuss platform future with partners | [L5,3] | E3, E9 |
| | **Manage ecosystem architecture** | A31 | Expand technical possibilities of the platform | [L13,9,3] | E2, E3, E6, E7 |
| | | A32 | Create technology that assists in app development | [L5] | E6*, E10 |
| | | A33 | Outsource core ecosystem functionalities | [L4] | E3, E9* |
| | | A34 | Assist partners in platform updates | [L5] | E9, E10* |
| | **Manage ecosystem knowledge** | A35 | Create a centralized repository of knowledge | [L4,3] | E3, E8, E9 |
| | | A36 | Share knowledge for ecosystem participation | [L13,5] | E2, E8*, E10* |
| | | A37 | Encourage knowledge sharing | [L4,12] | E3*, E8, E9 |
| | | A38 | Build industry consortium | [L7] | E3, E6, E7, E8 |
| Prosperity | **Encourage partner investments** | A39 | Define entry requirements | [L4] | E6*, E9 |
| | | A40 | Stimulate partner investments | [L12] | E4, E9 |
| | **Create associate model** | A41 | Differentiate Partners | [L12,6] | E5, E9 |
| | | A42 | Create partnership programs | [L12,5] | E5, E8, E10* |
| | **Align with other ecosystems** | A43 | Imitate competing platforms | [L13] | E5, E7 |
| | | A44 | Align with complementing ecosystems | [L9] | E7 |
| | **Promote innovation** | A45 | Promote adoption of key technology and standards | [L7,5] | E6, E8*, E9, E10 |
| | | A46 | Share innovations with partners | [L3] | E3, E8* |
| | | A47 | Support the protection of IP Rights | [L1] | E2, E3, E8 |
| | | A48 | Co-develop with partners | [L4,7] | E3, E8 |

Table 6: Partner management activities in software ecosystems. The descriptions and indicators of activities can be found in the Appendix. (*) Indicates that the enabler-activity relationships were not found in theory but observed in case studies.

mobile SECOs can certainly gain insights from the SECO-PMF, we do not claim to cover the complete set of PMAs these organizations employ.

Finally, we must frame the concepts that are addressed in this work. We mostly follow the theoretical framework outlined by Wouters et al. [78], who have made a first attempt at creating a vocabulary for SECOs. Also, we take a socio-technical resource view on orchestrators [46], but insufficiently reflect on the resources of partners to call it a full evaluation of those.

## 4. Partner management in the accounting software market

In this section, we evaluate the SECO-PMF's completeness and applicability by using it to assess the practices of the Dutch SPOs. We first describe the case study context and the relevance of this market for partner management, and then describe partner management in the accounting software industry in five ecosystems.

### *4.1. Industry Context Description*

The accounting software industry in the Netherlands finds its origins in the traditional accounting industry. The use of IT for accounting provides numerous benefits over paper ledgers. Even though accounting software serves varying markets, the products fulfill the same need for all customers, namely, semi-automated bookkeeping. We limited ourselves to the Dutch accounting software industry, to allow comparison between companies that serve the same market.

There are two types of accounting software, tailor-made software and product software [12]. Tailor-made accounting software requires investments that few companies have available. Therefore, companies that have not invested in tailor-made software use an accounting software product that is not developed specifically for that company.

The software producers attract and maintain partnerships with resellers, service providers, and software vendors, to use their expertise in attracting customers. The partnerships with software vendors enable an accounting software producing company to meet complex and diverse customer needs. We identify three types of extensions that the accounting software product benefits from:

- **Horizontal extensions** connect obligatory features for all customers that the software producer cannot make itself. An example of such a connection is forming a partnership with banks, as that enables customers to access their bank details, which accounting software producers cannot do themselves.
- **Vertical extensions** support the company's general customer base. Time tracking administration or payroll administration are features that may not be in the core platform but that interest customers and attracting partners who develop such extensions enables the accounting software company to remain focused. In such a distribution of labor, every company can invest in its own duties, thus improving the overall value distribution.
- **Niche integrations** satisfy a specific customer group's needs. For example, farmers require an accounting software package that tracks food intake, milk supply, and medicine intake of cattle. These extensions are irrelevant for non-agriculture customers. Therefore, partnering with companies that possess this knowledge enables the accounting software firm to meet farmer needs without investing themselves.

The accounting software ecosystem in the Netherlands consists of around 60 accounting software producing firms that compete for customers, according to an industry report [13]. The firms are central actors that form connections between partners and customers with the intent to nurture an ecosystem and platform surrounding a product. Therefore, the accounting software firms can be seen as orchestrators in the accounting software market. In general, these companies face intense competition due to product homogeneity and the number of orchestrators. The orchestrators, therefore, diversify their products to address a specific market niche, such as freelancers or multinationals, or a specific region, to avoid competing with the entire market [18].

Next to the orchestrator, several hundreds of other actors produce modules or extensions that satisfy accounting software customers. Both single-homing and multi-homing complementor strategies are present in the ecosystems [34]. Single-homing partners provide an ecosystem with an exclusive complementary product over rivaling ecosystems. Other partners are multi-homing, and offer their services to more customers in this way, while stripping orchestrators of a potential exclusive advantage. Within an ecosystem of an accounting software firm, the orchestrator may not be the largest or most influential company. Partners such as banks participate in a software ecosystem and may be more influential than the orchestrator is. Such an actor can exert power over orchestrators and their ecosystem [63].

Not every vendor of accounting software has managed to or is willing to grow a software ecosystem. Some companies attract fewer than 10 partners, where the largest app store contains around 750 extensions from 500 partners. Company E is a market leader as it has the largest number of customers, revenue, and number of partners of the case studies.

### *4.2. Partner Management Activities in Practice*

The five companies manage their partners differently. They each perform a custom set of activities. We describe the activities that they perform, and the activities that they do not perform in Table 7. If an activity is not performed, the motivation for not performing it is identified. We identified three types of motivation for not conducting an activity:

1. **Budgeting constraint**. The company lacks resources to carry out this activity (mentioned 33 times). The company recognizes the relevance and benefits of doing the partner management activity but is constrained by resources or budget.
2. **Unhelpful**. The activity does not have any perceived benefits for the orchestrator (mentioned 87 times). The company does realize that the activity can benefit partners, but does not identify benefit for the orchestrator.
3. **Irrelevant**. The activity is not relevant in this ecosystem (mentioned 14 times). The orchestrator does not perform activity, as they do not recognize benefits for partners.

The most frequently mentioned motivation for not carrying out an activity is that there are no benefits for the orchestrator. Therefore, partner management appears mature, as orchestrators carry out the activities that are beneficial for partners and do not perform redundant activities. However, one orchestrator may find value in activities that another orchestrator does not see benefits for.

It appears that within this market, orchestrators have specialized in a niche, where certain activities are more relevant or irrelevant. None of the organizations share innovations with partners, even though there is a general trend towards open innovation in ecosystems, especially in mobile ecosystems [23]. We hypothesize that under a certain ecosystem and orchestrator size, it is simply unimaginable to share intellectual property openly with others.

Another interesting observation is that companies A and B actively "avoid orchestrator incursion by contracts", i.e., building software into the platform that has already been built by the contracted partner, whereas the other three do not. We hypothesize that it is a strategic choice to openly commit to not rebuilding features into the platform that are already supplied by partners, which in turn may be determined by the amount of power an orchestrator can exert over its partners [63].

We combine the results of Table 7 in enabler scores and partner goal scores in Table 8. We compare the five orchestrating companies on the prevalence of enablers and the degree to which their ecosystem enables satisfaction of partner goals. A score of 100 for an enabler indicates that the orchestrator performs all activities that establish that ecosystem enabler. A score of 50 indicates that the orchestrator executes half of the activities that positively influence the ecosystem enabler. Similar to this, we calculate partner goal score by averaging the ecosystem enabler scores that lead to that partner goal. While it is hard to compare differently sized practices to each other, such as "certify partner applications" versus "create an app store", this indicative benchmark table provides rapid insight into the partner management practice maturity at these firms.

| Partner Mgmt Area | | Partner Management Activity | Case A | | Case B | | Case C | | Case D | | Case E | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | Used | Mot. | Used | Mot. | Used | Mot. | Used | Mot. | Used | Mot. |
| Create revenue models | A1 | Create revenue model | Yes | | No | 2 | No | 3 | Yes | | Yes | |
| | A2 | Enforce revenue streams | Yes | | Yes | | Yes | | Yes | | Yes | |
| Define quality standards | A3 | Create partner certificates | Yes | | No | 2 | No | 2 | No | 2 | Yes | |
| | A4 | Certify partner applications | No | 2 | Yes | | Yes | | Yes | | Yes | |
| Define rules to manage relationships | A5 | Create ecosystem principles | No | 2 | No | 2 | No | 2 | No | 2 | No | 2 |
| | A6 | Avoid orchestrator incursion by contracts | Yes | | Yes | | No | 2 | No | 2 | No | 2 |
| Enable effective communication channels | A7 | Set up personal partner contact | Yes | | Yes | | Yes | | Yes | | Yes | |
| | A8 | Develop ecosystem communication channels | No | 1 | No | 1 | No | 2 | No | 1 | Yes | |
| Attract and maintain partners | A9 | Create new partnerships | Yes | | Yes | | Yes | | Yes | | Yes | |
| | A10 | Discuss ecosystem participation with partners | Yes | | No | 2 | No | 2 | No | 2 | Yes | |
| | A11 | Support nontechnical partners | No | 1 | No | 3 | No | 2 | Yes | | Yes | |
| Create ecosystem community | A12 | Create a community forum | No | 2 | No | 2 | Yes | | No | 2 | Yes | |
| | A13 | Create partner hubs | No | 2 | No | 2 | Yes | | No | 1 | No | 2 |
| | A14 | Organize conferences | Yes | | Yes | | Yes | | Yes | | Yes | |
| Nurture collaborations | A15 | Train partners | No | 1 | No | 1 | Yes | | No | 2 | Yes | |
| | A16 | Create shared R&D teams | No | 1 | No | 2 | No | 2 | No | 1 | No | 3 |
| | A17 | Solve conflicts between partners | No | 1 | No | 2 | No | 2 | No | 2 | No | 2 |
| | A18 | Point out excelling partners | Yes | | Yes | | No | 2 | No | 2 | Yes | |
| Support in marketing and sales | A19 | Share sales knowledge and channels | No | 1 | No | 2 | Yes | | No | 2 | Yes | |
| | A20 | Share marketing channels | Yes | | No | 2 | Yes | | Yes | | Yes | |
| | A21 | Open platform to diverse markets | Yes | | Yes | | Yes | | No | 1 | Yes | |
| | A22 | Create joint marketing agreements | No | 2 | Yes | | Yes | | Yes | | Yes | |
| | A23 | Create an app store | Yes | | Yes | | No | 2 | Yes | | Yes | |
| | A24 | Recommend partners to customers | Yes | | Yes | | Yes | | Yes | | Yes | |
| | A25 | Bundle third-party applications and platform | No | 2 | No | 2 | Yes | | Yes | | Yes | |
| | A26 | Create public recognition for the ecosystem | Yes | | No | 3 | Yes | | No | 2 | Yes | |
| | A27 | Share reputation with partners | Yes | | No | 3 | No | 2 | Yes | | Yes | |
| Share roadmaps | A28 | Create ecosystem roadmap | No | 2 | No | 2 | No | 2 | No | 2 | Yes | |
| | A29 | Align partner strategies with roadmap | No | 2 | No | 3 | No | 2 | No | 1 | Yes | |
| | A30 | Discuss future platform technology with partners | Yes | | Yes | | No | 2 | No | 1 | Yes | |
| Manage ecosystem architecture | A31 | Expand technical possibilities of the platform | Yes | | Yes | | Yes | | Yes | | Yes | |
| | A32 | Create technology for app development | No | 2 | Yes | | No | 2 | Yes | | Yes | |
| | A33 | Outsource core ecosystem functionalities | No | 1 | No | 1 | No | 2 | No | 2 | Yes | |
| | A34 | Assist partners in platform updates | No | 2 | No | 2 | Yes | | No | 2 | Yes | |
| Manage ecosystem knowledge | A35 | Create a centralized repository of knowledge | No | 1 | No | 2 | No | 2 | No | 1 | Yes | |
| | A36 | Share knowledge for ecosystem participation | Yes | | Yes | | Yes | | Yes | | Yes | |
| | A37 | Encourage knowledge sharing | No | 1 | No | 2 | No | 2 | No | 1 | Yes | |
| | A38 | Build industry consortium | No | 1 | No | 2 | No | 2 | No | 1 | No | 1 |
| Encourage partner investments | A39 | Define entry requirements | No | 3 | No | 2 | No | 2 | No | 3 | No | 2 |
| | A40 | Stimulate partner investments | No | 2 | No | 2 | No | 2 | No | 3 | No | 2 |
| Create associate model | A41 | Differentiate Partners | No | 1 | No | 2 | No | 2 | Yes | | Yes | |
| | A42 | Create partnership programs | No | 1 | No | 2 | No | 2 | Yes | | Yes | |
| Align with other ecosystems | A43 | Imitate competing platforms | No | 2 | No | 2 | No | 2 | No | 3 | Yes | |
| | A44 | Align with complementing ecosystems | No | 2 | No | 3 | No | 3 | No | 3 | No | 3 |
| Promote innovation | A45 | Promote adoption of key technology and standards | No | 1 | No | 1 | No | 2 | No | 2 | Yes | |
| | A46 | Share innovations with partners | No | 1 | No | 1 | No | 2 | No | 2 | No | 2 |
| | A47 | Support the protection of IP Rights | No | 1 | No | 1 | No | 2 | No | 2 | No | 2 |
| | A48 | Co-develop with partners | No | 1 | No | 1 | No | 2 | Yes | | No | 2 |
| | | **Total number of performed activities** | **18/48** | | **15/48** | | **18/48** | | **19/48** | | **36/48** | |

Table 7: Practiced activities per case study. We list whether the activities are performed (Yes/No), and the identified motivation (Mot.) if it is not implemented. Explanation of the motivation types (1, 2 or 3) can be found in Section 4.2.

## 5. Partner Management Analysis and SECO-PMF Evaluation

In this section we explore the case study results per PMA. We complement the data of Table 7 with interview quotes. The researchers translated the quotes from Dutch to English. We highlight 8 out of 15 PMAs that characterize partner management in this market. We added quotes that form insight in partner management. The other PMAs provided trivial findings and have not been added to the analysis.

### *5.1. Create Revenue Models*

Three of the five companies have established a revenue model for their software ecosystem. There are different revenue models in use. Company B uses a revenue model where revenue streams are separate. The other orchestrators do it differently, for instance by taking over billing for partners. Customers pay both the platform price, as well as the extension price to the orchestrator, who then transfers the extension revenue to partners.

| Ecosystem Enabler | SPO E | SPO C | SPO D | SPO A | SPO B | Average |
|---|---|---|---|---|---|---|
| E5 Perceived product quality | 100 | 56 | 56 | 56 | 44 | **62** |
| E1 Personal and geographical proximity | 80 | 80 | 40 | 60 | 40 | **60** |
| E2 Mutual trust | 82 | 47 | 41 | 53 | 35 | **52** |
| E10 Lifecycle | 92 | 38 | 54 | 31 | 23 | **48** |
| E4 Effective communication | 67 | 33 | 33 | 42 | 25 | **40** |
| E3 Openness for technical and business negotiation | 70 | 35 | 39 | 30 | 22 | **39** |
| E8 Knowledge absorption | 67 | 39 | 33 | 22 | 17 | **36** |
| E9 Orchestrator leadership | 74 | 32 | 21 | 26 | 21 | **35** |
| E6 Availability of internal standards | 73 | 18 | 36 | 27 | 18 | **33** |
| E7 External alignment | 50 | 25 | 25 | 25 | 25 | **30** |
| **Total ecosystem score** | **75** | **40** | **38** | **37** | **27** | |

| Partner Goal | SPO E | SPO C | SPO D | SPO A | SPO B | Average |
|---|---|---|---|---|---|---|
| G4 Business expansion | 79 | 48 | 42 | 41 | 31 | **48** |
| G3 Platform integration | 79 | 43 | 41 | 43 | 28 | **47** |
| G1 Customer access | 80 | 38 | 39 | 39 | 28 | **45** |
| G2 Customer demand | 70 | 31 | 35 | 29 | 22 | **37** |
| G5 Cost reduction | 64 | 30 | 28 | 27 | 23 | **34** |
| **Total ecosystem score** | **74** | **38** | **37** | **36** | **26** | |

Table 8: Enabler scores and partner goal scores for the five case study companies. The SPOs are listed are listed left to right on highest score across enablers. The enablers and partner goals are listed top to bottom on highest score across ecosystems.

> "*We now use an API fee. We want 10% of the entry level product price. It is difficult because every partner has a different price model, so accessing APIs costs differently for every partner. However, this solution is per partner and not scalable, so we are looking for a revenue model where we say: this API has this price, and this API has this price.*" – Company E

The billing is paid for through an API fee in the ecosystem of company E. This is currently set per partner, but a long term scalable solution would be to set a price for partners per accessed API. The other ecosystems do not have a price for accessing the APIs, but employ a fair use policy. The difference in revenue models may be attributed to platform maturity, as mature orchestrators may have elaborate revenue models. Earnings flow from customers to the orchestrator who divides it among stakeholders. This allows the orchestrator to control partners and form dependency on the ecosystem. In the other ecosystems, stakeholders interact with the customer independently for billing.

> "*There are no revenue streams between the partners and our company. The business models are separate and everybody carries their own weight.*" – Company B

There is a difference in degree of control that the orchestrator exerts over partners in revenue models. Revenue models can be open, where every company determines its own business model (billing, pricing), or the orchestrator takes control over billing and pricing for partners. The restrictive revenue model is in line with a walled garden strategy, where the orchestrator sets strict rules for ecosystem growth [6,74].

The software ecosystem business model determines the entry barriers for complementors to join an ecosystem. Developer fees, SDK licenses, and app store commissions are reliable ways to get commitment and revenue from partners, but also ways to raise ecosystem entry barriers for extension developers [65]. We hypothesize that larger orchestrators have more power over their partners and have more resources to form advanced revenue models for their partners. **After a particular number of partners have become dependent on a platform, it becomes possible and profitable to develop commercial partnership models.**

### *5.2. Define Rules to Manage Relationships*

None of the five companies defined ecosystem principles for relationships, as it is perceived as unhelpful. Two of the orchestrators did have contracts in place to avoid orchestrator incursion. Orchestrator incursion is defined as partner market share takeover by the orchestrator. This leads to conflicts in the ecosystem, as partners feel threatened. Incursion can only occur when there is technical overlap between the platform and a partner's extension. An orchestrator can then ban the partner's extension from the ecosystem and force migration to the platform. Even though incursion appears profitable for the orchestrator, it negatively

affects the relations with partners who no longer feel supported by the orchestrator. Incursion is handled differently by the orchestrators, as the following quotes describe:

> "*This incursion is something that will always happen. If you have a large customer base, we will probably incorporate you instead of copying your product. If you refuse our takeover bid, we will come with counter measures. We set API fees or API gateways.*" – Company E

> "*There may be overlap between the platform's functionalities and the functionalities of extensions. We keep extending the platform based on the road map, so partners are aware that they have redundant extensions. We see customers migrating from the partner to the product functionality of the platform, but we do not actively subvert our partners. We even turn down some companies for a partnership, as their products overlap with the platform.*" – Company A

Company A recognizes the functionality overlap between platform and extension, but, to abstain from negatively affecting partners' trust, does not "steal away" customers. To avoid future conflicts, company A refuses partnerships that could lead to incursion at an early stage of collaboration. Company E uses a different approach, made possible by its larger size. Company E also recognizes the prevalence of incursion, but actively steers towards acquiring partner's customers. While this may threaten partners whose market share may also be taken over, the orchestrator acquires customers. We observe that orchestrator incursion is a phenomenon in this market that is handled differently by the SPOs. **The orchestrators can prevent overlapping platform extenders from entering the ecosystem, or take over partners that thrive in the ecosystem.** The other activities of "define rules to manage relationships" have remained implicit.

### *5.3. Attract and Maintain Partners*

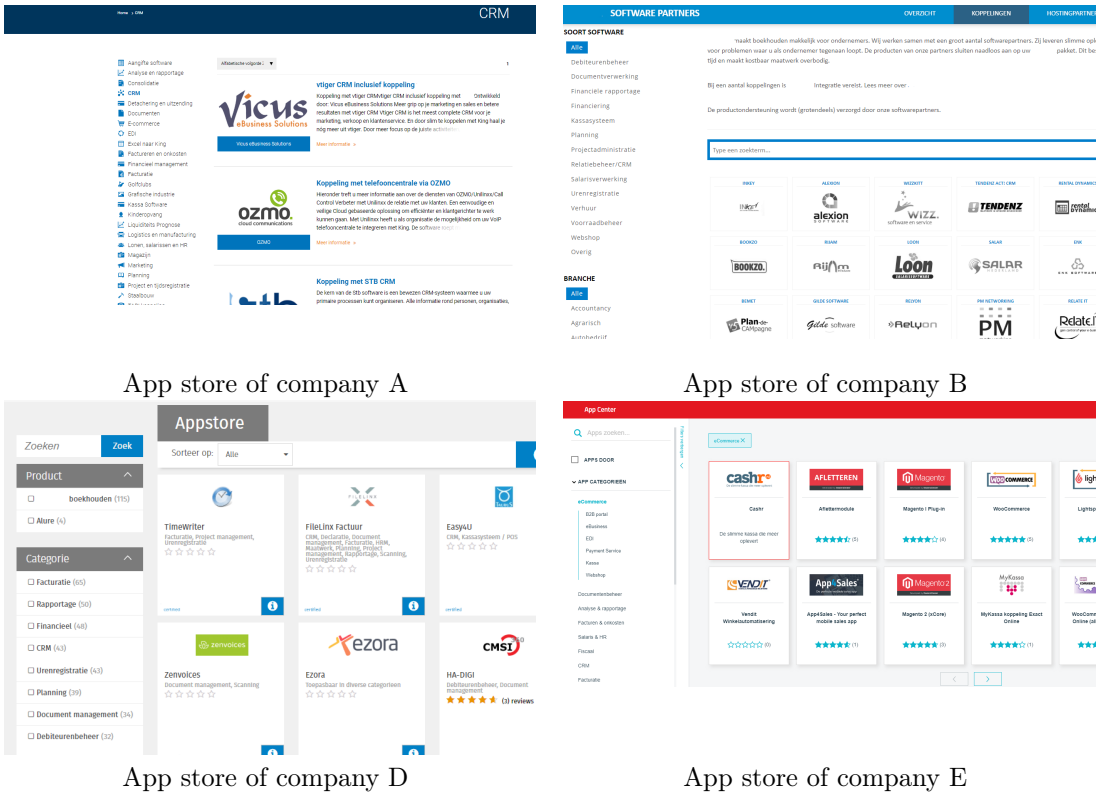
The orchestrators strive for an ecosystem that can meet the requests of customers, but that also has sustainable prospects. The platform provider encourages new partnerships, as this helps the ecosystem grow. simultaneously, the orchestrators avoid competition, both between partners and between the platform and the partners. Company A avoids this competition by excluding partners that may take over customers. Refusing access to the platform reduces the competition between the platform and the competitor's extension.

> "*We may exclude partners from the platform if their products do not match our platform vision. Our partners should address the same customer groups but should not be competing with our platform.*" – Company A

Company D describes a specific type of partnership that is not beneficial to the orchestrator. The partnership appears to be a prerequisite to an accounting software ecosystem, as customers turn away if they cannot access the extension of that partner. This type of partnership does not provide an advantage over other ecosystems, but it does provide a drawback if the orchestrator does not provide a partnership with this partner. This relation has a different power dynamic than other orchestrator-partner relations. The orchestrator does not see business value in collaborating with the partner, but, since its customers request the extension, the company is forced to form a partnership with this partner.

The customer-demanded partner appears to have established a form of power over the orchestrators. The orchestrators may be more exchangeable than that partner is. This leads to a situation where the partner can make business proposals that are favorable to itself and not to the orchestrator. Thus, company D does not benefit from the partnership with the prerequisite partner.

> "*We have a partner that connects to all our competitors. It provides a set of data analytics for our customers. However, the partner does not add any additional value to our platform, and it uses a lot of our bandwidth. That partner is a parasite to our product, as maintaining the connection costs money, and we make nothing from it. But we cannot repel this partner, as our customers will start complaining. It indicates that we cannot determine who our partners are; our customers do so.*" – Company D




App store of company A | App store of company B

App store of company D | App store of company E

Figure 4: Screenshots of case company app stores as part of our evidence collected. Four out of the five case companies (A, B, D, and E) use an application store for partners to present their solutions, which was identified as one of the partner management activities.

In ecosystem orchestration, the orchestrator can predefine partner requirements, and therefore affect which actors join the ecosystem. For example, Microsoft attracts many partners of varying sizes, while SAP attracts fewer and larger partners than Microsoft [36]. **Even when the orchestrator sets expectations for partners, customers ultimately determine which partners stay in the ecosystem.** This is in line with Riehle, who observed that customer's request and the number of customers determine the feasibility of partnerships [54].

### *5.4. Support in Marketing and Sales*

Orchestrators vary in supporting partners in marketing and sales. Company C, D, and E orchestrators aid in sales and marketing of partners, while company A and B do not. The approach in offered support varies as well. Orchestrator B does some personal face-to-face recommendation, while the company E performs extensive shared marketing campaigns with partners.

> "*We use "in-product" marketing messages to customers. When a customer does not use a certain product that similar customers do use, he receives advertisements for that extension. We include it in the API fee, and partners love it.*" – Company E

Four out of the five orchestrators use an app store to market partner applications. We provided a screenshot of every app store in Figure 4. The app stores all categorize apps by functionality. One orchestrator does not have an app store. In that ecosystem, customers find the extensions by contacting the orchestrator with a specific request, instead of finding the extension in the app store.

Company E possesses an interface that provides personalized advertisements to customers, which the other companies do not have. This helps the partners in sales and tightens the relationship between orchestrator and partner. Company E uses the "in-product" advertising as a selling point to encourage potential

partners to join the ecosystem. The orchestrator can recognize a customer's unsatisfied demand, and suggest which partner extension meets the demand [17].

> "*There have been sessions with partners for combined marketing. But our marketing and sales always hammer on the number of leads we get ourselves, not for our partners. That makes it difficult to convince marketing to work for partners.*" – Company E

We observe that there is a clash between internal marketing goals and ecosystem goals. Even though successful partner management allows the orchestrator to benefit from thriving partners, the marketing team focuses on achieving leads and customers for the orchestrator only. Iyer et al. found the benefits of ecosystem marketing and sales but the difficulty of those departments to pursue ecosystem goals [36]. Ecosystem marketing can be encouraged by collaborating with partners that sell to the same customers group, at the same technical layer [L3]. **Therefore, selecting the right partner for marketing collaboration is required to convince the marketing team to help the partner.**

### *5.5. Manage Ecosystem Architecture*

One company successfully outsources a core platform functionality. Company E has left development of the invoice matching function, i.e., the matching of invoices to entities in the customer relationship management portion of the system, to a partner, which enables the company to stay focused and spend more resources on orchestrating. Smaller companies appear to lack the funds and interest in sharing platform development. Perhaps the size of company E and its ecosystem makes the company recognize that orchestrating and outsourcing is more efficient than developing the entire platform itself. Microsoft outsources parts of its core ecosystem, including the training of partners [10].

> "*It was difficult to develop an invoice matching feature in our product. After matching became popular, a lot of matching applications grew immensely. We felt it was our social obligation to develop a matching feature and connect with the large number of banks and payment service providers. However, it would take a long time to connect to all companies, so we ended up partnering with a company. That company developed our matching product.*" – Company E

The partner community can take over tasks from the SPO. Suitable outsourcing tasks differ per ecosystem. The partner community of SAP supports in pre-sales activities, sales activities, product distribution and support [52]. **Outsourcing orchestrating activities results in costs savings.**

### *5.6. Encourage Partner Investments*

Orchestrators do not encourage additional partner investments, as they perceive no benefit to the ecosystem. It appears as if dependency and loyalty to the ecosystem is of little concern for orchestrators. From their perspective, a partner extends what the platform can offer customers. A thriving ecosystem, where partners interact or where partners are encouraged to invest in the ecosystem, is not the platform provider's concern.

This could be explained by the platform's nature. As the interviewee states, the platform is a commodity product, and partners can substitute one platform for another. We observe that this does not lead to the tightened and trusted relations that occur in ecosystems where platform substitution is impossible. If partners can change ecosystems easily, there is no incentive to invest in partner loyalty. Contrasting, in gaming software ecosystems, game developers can decide on which platform they release games. In these ecosystems, achieving partner loyalty and being able to sell partners' games is essential for launching consoles and survival of ecosystems [75]. We observe that **orchestrators benefit from investing in sustainable relations with partners when the ecosystem faces competition from other ecosystems and partners can become dependent on the ecosystem**.

> "*I do not see any partner loyalty. If [Company D] goes bankrupt tomorrow, all our partners are gone. [...] Partners are not dependent on us; our platform is a commodity product.*" – Company D

*5.7. Create Associate Model*

The lead companies distinguish different partnerships in their ecosystem. The orchestrators maintain relationships with groups of independent software vendors and system integrators. Company A, B, and C do not have a partnership program. These companies may not have sufficient resources to invest in partners. The ecosystems of company D and E possess different levels of partner commitment. West and Wood suggest that two to four levels of partner importance should be used [75].

> "*We have gold, silver and bronze partners, and certain APIs are only open for those certified partners.*" – Company E

The interviewee of Company E describes how commitment affects the way a partner can interact with the ecosystem. **Creating value for partners that participate in a partnership program encourages partners to become a certified partner.** An orchestrator could assign other benefits, such as support, access to education, or coaching [1].

*5.8. Promote Innovation*

Regarding the promotion of innovation in an ecosystem, we found that one orchestrator promotes adoption of key technologies and related standards. Company E can promote a standard in the ecosystem, which benefits the ecosystem. Other orchestrators appear to lack the power to convince partners to adopt such a standard, as they cannot enforce standards in their ecosystem.

> "*We used to have a lot of partners connecting with an outdated XML standard. In our app store, we branded those partners as "potentially unsafe". It is over quickly then. Now, all those partners have a certified API connection.*" – Company E

One company actively co-develops with partners. Company D has jointly developed an application. The intense form of collaboration appears to be beneficial for both parties. The partner provides agility and creativity and the orchestrator can use its branding to form a sustainable partnership. Bridging the gap between the leading company's resources and the agility of the nascent company brings profits to both companies [73].

> "*For valued partners, there is a new opportunity of co-marketing products. In this intensive form of cooperation, we create a single brochure, website, and even have a shared corporate design. We discuss and share all marketing aspects; the sales kit, target audience, and the story. So far, this collaboration has been a great success and we are willing to extend this to other partnerships. We can combine the small partner's agility with the orchestrator's reputation.*" – Company D

We observe that the SPOs do not carry out the activities related to the promotion of innovation. Sharing innovations or supporting IPR require large-scale investments that can only be performed in tight relations. However, in this market, we observed that there is no partner loyalty. **The absence of partner loyalty reduces the opportunities to co-innovate**. Furthermore, established protection of IPR is required for collaboration in innovation [L1]. Since the SPOs have not established IPR protection regulations, partners may not be open for collaboration.

*5.9. Manage Conflicts*

None of the partners stated that they manage conflicts between partners. It is interesting to investigate why the orchestrators do not solve conflicts. Perhaps there are no conflicts at all in the ecosystems. However, we doubt this, as the ecosystems contain partners that develop competing applications. Two partners that compete for market share could generate conflicts. It may also be possible that there are conflicts between partners, but the orchestrators do not see it as their duty to solve these conflicts. In this case, one could evaluate the benefits and drawbacks of stepping in and managing the conflicts between partners. Benefits include recognition of the orchestrator's authority and harmony in the ecosystem. Drawbacks include the establishment of the orchestrator as a despot, and partner dislike such an intrusive leader.

## 6. Framework Adaptation

The evaluation of a model by using case studies may miss elements that are not present in the model [80]. To measure the completeness and validity of the SECO-PMF, we asked the interviewed industry practitioners open-ended questions regarding their perceived framework validity and completeness.[10] Next to the practitioners, we consulted four academic experts for their assessment of the SECO-PMF. The four experts are experienced researchers in the software ecosystems domain. The model validity will be discussed in Section 8. In this section, we describe how the framework was refined based on the practitioners' and academic experts' evaluations.

We gathered the feedback of academic experts through interviews. We conducted a step-by-step investigation of the SECO-PMF: for every element (Partner Goal, Enabler, PMA and Activity), we asked how it matched the academic expert's perception and experience of partner management in software ecosystems. We concluded the interview by asking whether the expert had any other framework recommendations. The framework adaptation remarks, defined as suggestions to modify elements of the framework, were noted on paper. This approach follows the Design Science guidelines of Hevner and Chatterjee [32].

Overall, both the practitioners and the academic experts emphasized the relevance of creating a classification of partner management activities. In total, the evaluations resulted in 29 framework adaptation remarks.[11]

Five remarks originated in the practitioner evaluations and 24 came from the academic expert evaluations. The remarks can be categorized as follows:

1. Thirteen remarks suggested the merge of PMAs or activities. The academic experts explained that the theoretical difference between activities would not manifest itself in practice. Therefore, they suggested a merge of the elements in the framework. For example, *Share sales knowledge with partners* and *Share sales channels* were merged, based on expert suggestion;
2. Ten remarks are suggestions to improve the clarity of elements. We agree that the framework's short sentence structure may lead to uncertainty regarding the activities' meaning. For example, *Assist partners in developing* was renamed into *Support non-technical partners in accessing the ecosystem*, as an expert remarked that it would improve clarity of the activity;
3. Four new activities have been suggested. These are added to the SECO-PMF;
4. The other remarks concerned elements that should be moved to other sections of the framework.

All remarks suggested by the practitioners and experts have been used to refine the framework.[12] The result of these refinements is the final version of the SECO-PMF, as visualized in Figure 3 and written out in Table 4 (Partner Goals), Table 5 (Enablers), Table 6 (Activities) and the Appendix (Activity Descriptions). We have chosen to only insert the final version of the framework in this article, to limit the article's length and prevent confusion regarding the final version.[13]

## 7. Research Contributions

In this research, partner management has been investigated from a theoretical and practical perspective. This has resulted in the SECO-PMF that classifies the aspects of managing partners. The SECO-PMF aids orchestrators in understanding the effects of ecosystem governance. In this section, we name the relevant contributions to research and practitioners.

[10] The questions for this evaluation can be found in the interview protocol, footnoted in the Research Method section

[11] The list of framework adaptation remarks is accessible online at Mendeley data [67].

[12] The interpretation of the remarks and commentary on how processing the remarks changed the framework can be found in the dataset, ibid.

[13] the version before the framework adoption is available in the "SECO-PMF Adaption" dataset, ibid.

### *7.1. Scientific Contributions*

We list the two contributions that this article makes to ecosystem literature:

**1. Partner management overview by categorizing partner management activities and enabling reasoning about the partner management domain as a whole.** Partner management in ecosystems consists of a diversity of activities. The articles used as literature set investigate the individual partner management activities.

However, these articles concern individual activities and not a unified governance mechanisms categorization. To the researchers' knowledge, no classification of knowledge and activities regarding partner management has been established. Therefore, this research defined activities, ecosystem enablers and partner goals to aid the scientific debate, created categories for the activities in the literature, and proposes the SECO-PMF as an exhaustive list of partner management activities. These contributions enable software ecosystem researchers to step beyond individual partner management activities and reason about the entire domain.

**2. A comprehensive perspective to assess partner management practices.** Literature that concerns partner management takes the perspective of investigating the effect of a single or multiple activities. Although this approach offers in-depth understanding of a single activity's consequences, it does not enable understanding of partner well-being, as partners are simultaneously exposed to all orchestrator interventions. When investigating ecosystem health or the effectiveness of partner management, one cannot limit herself to a single activity. Activities interfere with each other or have contradictory effects. Only by measuring the effect of all activities, can a researcher examine if the ecosystem and partners perform as intended. To incorporate the comprehensive perspective that a concept can only be measured by investigating it as a whole, the SECO-PMF establishes understanding of the combined effects of all activities on the ecosystem and partners. This allows one to move beyond the effects of individual governance mechanisms and identify partner management strategies consisting of all activities that change the ecosystem in a specific direction.

### *7.2. Contributions for Practitioners*

The SECO-PMF provides an overview of the practices that are available to SPOs in managing their partner networks. The activities, as found in Table 6, are related to partner goals through ecosystem enablers, i.e., factors that enable partners to reach particular goals. The activity "Create partner hub", for instance, contributes to the ecosystem enabler "Orchestrator leadership", which in turn enables partners to reach the goal of customer access. For practitioners, such as business managers at SPOs, we provide three applications for a **Research-based methodology to assess ecosystems**, at different life cycle phases of the platform.

**1. Building blocks for Creating an Ecosystem**. In the early phases of a software platform, an orchestrator-to-be can benefit from the SECO-PMF while opening the platform for extensions and should think of the partner goals that the ecosystem should fulfill, as listed in Table 4. The orchestrator should investigate which ecosystem enablers the ecosystem should possess for partners to reach these partner goals, as indicated in Table 5. Finally, the orchestrator selects activities that positively influence enablers that it aims at, in Table 6. Using a roadmap, which is largely fed by the SECO-PMF, organizations can develop a timed overview of how they plan to implement their activities.

**2. Partner Management Process Assessment**. When an ecosystem has grown around a software platform, the orchestrator can use the SECO-PMF to form an elaborate evaluation of partner management and identify opportunities for improvement. Evaluating partner management can identify mismatches in activities and the intended goals of the ecosystem. If there are discrepancies between the intended partner management and the partner management practice, the ecosystem governance team can identify the activities that should be added or removed. Finally, the team should create and execute a roadmap for adding or removing those activities.

**3. Partner Management Peer Assessment**. When the SPO is orchestrating a significantly sized ecosystem, it can evaluate itself in a competing market where multiple ecosystems compete for market share. Comparing partner management of one's own company to that of competitors may explain why the other ecosystem is more successful, has different partners, or provides a different value proposition to

customers. Highlighting the differences in partner management between ecosystems provides opportunities to improve partner management. The management team should identify competing ecosystems that partner management will be compared to and gather data regarding the activities that the orchestrators carry out in the ecosystems in scope. The orchestrator assesses its own partner management either through the approach described above, or a more elementary approach. The orchestrator then compares the partner management of all ecosystems to the SECO-PMF. By highlighting the performed activities per ecosystem, the orchestrator identifies the differences in partner management. Comparing the differences in partner management to the difference in business success, partnerships, or value proposition may provide understanding about how to improve partner management. The orchestrator starts performing neglected activities that improve partner management and removes redundant activities.

A continuous performance measurement for partnerships is a management task that supports organizations in optimizing their resource distribution, i.e., how much time they should spend on each partner and on partner management processes in general [4,70]. The interviewees state that they fail to convince the management board of the effectiveness of partner management for many of the practices. Using the SECO-PMF is stepping beyond the black box that partner management is currently perceived as, and enables a measurement of the effectiveness of partner management. This helps in objectifying the results of ecosystem governance and can be used to convince the management team of the value of maintaining an ecosystem.

## 8. Discussion

In this section, a reflection is provided on the research process and results, addressing model validity and the case study approach.

The framework was evaluated in five case studies that share a context, i.e., the bookkeeping and accounting market. There are other ecosystems with different partner management activities than in the case studies, however, these were considered out of scope. Other ecosystems research projects also use a multiple case design [80], and compare between ecosystems [38,12]. In other ecosystems, other PMAs become more relevant. For example, in the communications industry, the orchestrator establishes software standards and a hardware platform [72]. In the case studies, the promotion of innovation by setting standards was not seen as the orchestrator's duty. Thus, an investigation of the broader case context is required to understand the relevance of PMAs.

### *8.1. Framework Discussion*

The SECO-PMF, aimed to be a generic framework, is not perfectly suited to partner management in any software ecosystem. By making the SECO-PMF applicable to multiple ecosystems, certain PMAs may become redundant in some ecosystems. For example, in the case studies, it became apparent that none of the orchestrators manages partner conflicts, even though in open source software ecosystems partner management concerns conflict management [L4]. There are ecosystem characteristics that cause the absence of those activities in the case studies. If the SECO-PMF were applied to, for example, a mobile application ecosystem, other activities would require more attention [7].

Partner management was investigated from the orchestrator's perspective and no time was invested in how partners respond to changes in partner management. An investigation of how partners respond to orchestrator activities may bring further understanding of why ecosystems thrive or collapse [64]. The SECO-PMF describes how an orchestrator should act to attract partners and support these partners. This is one of the aspects of ecosystem governance. However, other aspects of ecosystem governance are not covered by the SECO-PMF. To successfully govern an ecosystem, one should also investigate the platform's customers and have a strategy for profitability. These aspects are beyond the SECO-PMF's scope. This research contributes to understanding the management of partners, which is part of the quest for understanding the effect of ecosystems.

Other orchestrators can use the SECO-PMF as a frame of reference when creating a new platform, a new software ecosystem, and when implementing the partner management processes. Furthermore, the framework and data in this article can be used by orchestrators to benchmark themselves against the five

platforms. The SECO-PMF has only been evaluated in the bookkeeping domain in the Netherlands and we cannot generalize outside of this business to business ecosystem at this time. That said, we believe that many of the practices in the bookkeeping domain would also be useful for other software application domains.

### *8.2. Literature Study Reflection*

The creation of the SECO-PMF started with literature identification. In this process, we were confronted with validity threats regarding the construction of the search string and the selection of digital libraries. The construction of the search string impacts the relevance of returned results. To find results relevant to the research goal, we adopted concepts of the meta-model in the search strings. In a research area without a standard terminology such as ecosystems research, adding synonyms aids in finding relevant articles. Therefore, we added two synonyms for each concept in the search string.

Failing to select the right digital libraries leads to irrelevant studies or the miss of relevant studies [3]. In this research, the use of PoP software limited the available search engines to Scopus and Scholar. To avoid a search engine bias and to complement the existing set of articles, we added a forward and backward snowballing approach. This mitigated biases from using merely two search engines, as the access to articles is no longer dependent on the search engines used.

After running the search queries and searching forward and backward through the results, we are faced with validity threats of inclusion and exclusion bias. We attempted to select the articles relevant for model construction by setting strict exclusion criteria. After selecting the 13 articles that would populate the SECO-PMF, the model creation process commenced. This led to validity threats related to the data extraction. We created a standardized data extraction form and labeled the extracted concepts. In this way, the partner goals, enablers and partner management activities could be extracted in a structured manner. During the combination process, it became clear that the articles address partner management on different levels. To create a standardized list, we adapted the granularity level by splitting larger concepts and merging smaller concepts.

If we were to repeat this part of the project, we would improve on two parts. First, we would do a more extensive literature study with a larger team of researchers. This would probably lead to a larger set of articles, and possible a larger set of PMAs. Secondly, we would widen our exclusion criteria, to see if we could capture a larger set of literature to study in detail. This would contribute to comprehensiveness of the model.

### *8.3. Case Study Reflection*

In the case study, we identified that some PMAs and activities are carried out frequently, while others are not used at all. This indicates that not every PMA is equally important, and there is an order of importance when selecting new activities. For example, Wnuk et al. manages to capture the order of importance, causes, and effects of activities [L13]. However, in this research design, we could not establish the order of importance of PMAs. Instead, we categorized the activities according to the ecosystem lifecycle.

The SPOs' practices regarding the management of partners fit in the PMAs. It is found that the case studies do not require extension of the framework. Instead, investigating how an activity is carried out in practice contributes to practical knowledge of activities. We found that the case orchestrators sparsely fill the matrix with PMAs, i.e., they only fulfil a small part of the SECO-PMF. In Table 7 we illustrate why an orchestrator does not implement a particular practice. While this was not part of the original case protocol, it provides excellent insight into why orchestrators cannot implement practices.

### *8.4. Future Work*

The research work conducted for this study has been considerable, with interviews and five extensive case studies. However, some might consider this only to be the birth stage of the SECO-PMF. There are multiple directions we can take for future work, and these are discussed here shortly.

This research was limited to the accounting software ecosystem. Researchers that investigate other orchestrators who face different threats and opportunities and manage partners differently contributes to a deeper understanding of ecosystem health. Furthermore, examining entry barriers for partners remains a topic of interest, as these hinder ecosystem prosperity when partner management is successful.

Also, as the case studies only evaluate the completeness and applicability of the SECO-PMF, there is still a need to evaluate the ease of use and effectiveness of the SECO-PMF. Furthermore, we hope that ecosystems researchers contextualize partner practices in our framework and develop theory about the practices in detail.

If we were to extend the project, we would invest the time into performing several more case studies, with the goal of diversifying our data set and ensuring generalizability beyond the Dutch ERP market. We would initially not go beyond business to business software, but we would want to include multiple software companies in multiple countries to ensure that our work becomes generalizable to all orchestrators in the business to business industry. There are multiple types of case studies that we could potentially do in the future.

First, we could *perform more perfective case studies*, where the model is further adjusted based on its application in those case organizations. A relevant extension to this would be to include several partners of an organization in each case study to rate their perception of the PMAs executed by the orchestrator. This would provide a more comprehensive overview of the effectiveness of the activities proposed in the PMA. However, we will probably opt for *more confirmatory case studies* in different domains, where the PMF does not change, but we explore in a longitudinal manner whether the SECO-PMF and improvements stemming from it have improved partner performance and perhaps even measure the software ecosystem health [38].

One aspect that received little attention in the interviews and case studies is security of solutions provided by partners and the role of for instance security training of partners to provide the most secure extensions. In previous work, we have spent attention to this on the technical aspects of platform management [39], but security is increasingly becoming a concern in complex software supply chains and ecosystems. In the future, we intend to look at the various measures organizations take to protect their platforms and the solutions offered by partners.

Finally, an investigation is needed of the profitability of partner management. Investigating how an orchestrator can gather revenue by using partners and ecosystems will bring further understanding of the role of partner management in the success of orchestrators and the health of the software ecosystem.

## 9. Conclusion

Understanding the components partner management contributes to the orchestration of ecosystems. This article presents the SECO-PMF as a framework for the orchestration of relationships with partners. By designing and populating a meta-model from partner management literature, we created a framework that addressed partner management. The first version of the SECO-PMF was designed by clustering partner management activities from these articles. Subsequently, the SECO-PMF is further constructed and evaluated through an extensive multiple case study at five companies.

This article proposes The SECO-PMF to model partner management processes of software ecosystem orchestrators. In this framework, activities create ecosystem enablers that satisfy partner goals. This framework aids academics in three ways. To solve the uncertainty of the partner management components, we propose an activity classification in partner management activity. To aid in understanding the effects of partner management on an ecosystem, this article revealed how activities contribute to ecosystem enablers and goals. Finally, to solve the need for common vocabulary in ecosystems literature, activities, ecosystem enablers, and partner goals are proposed that describe partner management in a common vocabulary.

**Acknowledgements**
We thank the case study companies and the interviewees for granting us the time with them for our research. We thank the four academic experts for providing an evaluation of the SECO-PMF. Finally, we thank the four anonymous reviewers of the SCP journal, who provided us with numerous comments and helpful feedback that have significantly improved our work.

## Literature Set


[L1] M. Ceccagnoli, C. Forman, P. Huang, and D. Wu. Cocreation of value in a platform ecosystem: The case of enterprise software. *MIS quarterly*, 2012.

[L2] T. Hurni and T. Huber. The interplay of power and trust in platform ecosystems of the enterprise application software industry. *Twenty Second European Conference on Information Systems*, 2014.

[L3] T. Kude, J. Dibbern, and A. Heinzl. Why do complementors participate? an analysis of partnership networks in the enterprise software industry. *IEEE transactions on engineering management*, 59(2):250–265, 2012.

[L4] L. Morgan, J. Feller, and P. Finnegan. Exploring value networks: theorising the creation and capture of value with open source software. *European journal of information systems*, 22(5):569–588, 2013.

[L5] T. Rickmann, S. Wenzel, and K. Fischbach. Software ecosystem orchestration: the perspective of complementors. *Twentieth Americas Conference on Information Systems (AMCIS)*, 2014.

[L6] J. Rietveld, R. Seamans, and K. Meggiorin. Market orchestrators: The effect of platform certification on complementor performance and behavior in the context of kiva (2010-2013). *Available at SSRN 3131553*, 2018.

[L7] P. Ritala, V. Agouridas, D. Assimakopoulos, and O. Gies. Value creation and capture mechanisms in innovation ecosystems: a comparative case study. *International Journal of Technology Management*, 63(3-4):244–267, 2013.

[L8] M. Schreieck, M. Wiesche, and H. Krcmar. The platform owner's challenge to capture value - insights from a business-to-business it platform. In *38th International Conference on Information Systems*, 2017.

[L9] L. Selander, O. Henfridsson, and F. Svahn. Transforming ecosystem relationships in digital innovation. In *31st Inernational Conference on Information Systems (ICIS)*, volume 138, 2010.

[L10] C. Storey, C. Kocabasoglu-Hillmer, S. Roden, and K. de Ruyter. Governing embedded partner networks: certification and partner communities in the it sector. *11th International Journal of Operations & Production Management*, 2018.

[L11] G. Valença and C. Alves. Strategies to foster software ecosystem partnerships–a system dynamics analysis. In *International Conference on Enterprise Information Systems*, pages 189–211. Springer, 2017.

[L12] P. J. Williamson and A. De Meyer. Ecosystem advantage: How to successfully harness the power of partners. *California management review*, 55(1):24–46, 2012.

[L13] K. Wnuk, P. Runeson, M. Lantz, and O. Weijden. Bridges and barriers to hardware-dependent software ecosystem participation–a case study. *Information and Software Technology*, 56(11):1493–1507, 2014.


## References


[1] S. Ali and S. U. Khan. Software outsourcing partnership model: An evaluation framework for vendor organizations. *Journal of Systems and Software*, 117:402–425, 2016.

[2] C. Alves, J. Oliveira, and S. Jansen. Software ecosystems governance-a systematic literature review and research agenda. In *Proceedings of the 19th International Conference on Enterprise Information Systems*, volume 3, pages 26–29, 2017.

[3] A. Ampatzoglou, S. Bibi, P. Avgeriou, M. Ver-beek, and A. Chatzigeorgiou. Identifying, categorizing and mitigating threats to validity in software engineering secondary studies. *Information and Software Technology*, 2018.

[4] A. Avila and O. Terzidis. Management of partner ecosystems in the enterprise software industry. In *The 8th International Workshop on Software Ecosystems*, pages 39–55, 2016.

[5] O. Barbosa and C. Alves. A systematic mapping study on software ecosystems. *Third International Workshop on Software Ecosystems*, page 59–81, 2011.

[6] D. Barrera and P. Van Oorschot. Secure software installation on smartphones. *IEEE Security & Privacy*, 9(3):42–48, 2010.

[7] R. C. Basole. Visualization of interfirm relations in a converging mobile ecosystem. *Journal of information Technology*, 24(2):144–159, 2009.

[8] L. Beelen, S. Overbeek, and S. Jansen. Are you of value to me? a partner selection reference method for software ecosystem orchestrators. *Submitted*, 2021.

[9] Í. Belo and C. Alves. How to create a software ecosystem? a partnership meta-model and strategic patterns. *Information*, 12(6):240, 2021.

[10] E. Bonfiglioli, L. Moir, and V. Ambrosini. Developing the wider role of business in society: The experience of microsoft in developing training and supporting employability. *Corporate Governance: The international Journal of Business in Society*, 6(4):401–408, 2006.

[11] J. Bosch. From software product lines to software ecosystems. In *Proceedings of the 13th International Software Product Line Conference*, pages 111–119. Carnegie Mellon University, 2009.

[12] J. Bosch and P. Bosch-Sijtsema. From integration to composition: On the impact of software product lines, global development and ecosystems. *Journal of Systems and Software*, 83(1):67–76, 2010.

[13] G. Bottemanne. APIs en administratieve software: het nieuwe ERP. *Softwarepakketten.nl - Berichten - Raadplegen detail*, 2018.

[14] S. Brinkkemper, M. Saeki, and F. Harmsen. Meta-modelling based assembly techniques for situational method engineering. *Information Systems*, 24(3):209 – 228, 1999.

[15] C. Burkard, T. Widjaja, and P. Buxmann. Software ecosystems. *Business & Information Systems Engineering*, 4(1):41–44, 2012.

[16] M. Callon. Actor-network theory—the market test. *The Sociological Review*, 47(1):181–195, 1999.

[17] P.-T. Chen and H.-P. Hsieh. Personalized mobile advertising: Its key attributes, trends, and social impact. *Technological Forecasting and Social Change*, 79(3):543–557, 2012.
[18] M. G. Colombo, E. Piva, and C. Rossi-Lamastra. Open innovation and within-industry diversification in small and medium enterprises: The case of open source software firms. *Research Policy*, 43(5):891–902, 2014.
[19] M. Cusumano. *Staying power: Six enduring principles for managing strategy and innovation in an uncertain world (lessons from Microsoft, Apple, Intel, Google, Toyota and more)*. Oxford University Press, 2010.
[20] M. de Reuver, C. Sørensen, and R. C. Basole. The digital platform: a research agenda. *Journal of Information Technology*, 33(2):124–135, 2018.
[21] C. Dhanaraj and A. Parkhe. Orchestrating innovation networks. *Academy of management review*, 31(3):659–669, 2006.
[22] B. Eaton, S. Elaluf-Calderwood, C. Sorensen, and Y. Yoo. Distributed tuning of boundary resources: the case of apple's ios service system. *MIS Quarterly: Management Information Systems*, 39(1):217–243, 2015.
[23] J. T. Eckhardt, M. P. Ciuchta, and M. Carpenter. Open innovation, information, and entrepreneurship within platform ecosystems. *Strategic Entrepreneurship Journal*, 12(3):369–391, 2018.
[24] A. Gawer and M. A. Cusumano. Industry platforms and ecosystem innovation. *Journal of Product Innovation Management*, 31(3):417–433, 2014.
[25] A. Gawer and M. A. Cusumano. Platform leaders. *MIT Sloan management review*, pages 68–75, 2015.
[26] A. Ghazawneh and O. Henfridsson. Governing third-party development through platform boundary resources. In *the International Conference on Information Systems*, pages 1–18. AIS Electronic Library, 2010.
[27] A. Ghazawneh and O. Henfridsson. Balancing platform control and external contribution in third-party development: the boundary resources model. *Information Systems Journal*, 23(2):173–192, 2013.
[28] R. Gulati and D. Kletter. Shrinking core, expanding periphery: The relational architecture of high-performing organizations. *California Management Review*, 47(3):77–104, 2005.
[29] R. Gulati, N. Nohria, and A. Zaheer. Strategic networks. *Strategic Management Journal*, 21(3):203–215, 2000.
[30] E. Handoyo, S. Jansen, and S. Brinkkemper. Software ecosystem roles classification. In *International Conference of Software Business*, pages 212–216. Springer, 2013.
[31] A.-W. Harzing and S. Alakangas. Google scholar, scopus and the web of science: a longitudinal and cross-disciplinary comparison. *Scientometrics*, 106(2):787–804, 2016.
[32] A. Hevner and S. Chatterjee. Design science research in information systems. In *Design research in information systems*, pages 9–22. Springer, 2010.
[33] A. R. Hevner, S. T. March, J. Park, and S. Ram. Design science in information systems research. *MIS quarterly*, pages 75–105, 2004.
[34] S. Hyrynsalmi, A. Suominen, and M. Mäntymäki. The influence of developer multi-homing on competition between software ecosystems. *Journal of Systems and Software*, 111:119–127, 2016.
[35] M. Iansiti and R. Levien. *The keystone advantage: what the new dynamics of business ecosystems mean for strategy, innovation, and sustainability*. Harvard Business Press, 2004.
[36] B. Iyer, C.-H. Lee, and N. Venkatraman. Managing in a "small world ecosystem": Lessons from the software sector. *California Management Review*, 48(3):28–47, 2006.
[37] M. G. Jacobides, C. Cennamo, and A. Gawer. Towards a theory of ecosystems. *Strategic Management Journal*, 39(8):2255–2276, 2018.
[38] S. Jansen. Measuring the health of open source software ecosystems: Beyond the scope of project health. *Information and Software Technology*, 56(11):1508–1519, 2014.
[39] S. Jansen. A focus area maturity model for software ecosystem governance. *Information and Software Technology*, 118:106–219, 2020.
[40] S. Jansen and E. Bloemendal. Defining app stores: The role of curated marketplaces in software ecosystems. In *International Conference of Software Business*, pages 195–206. Springer, 2013.
[41] S. Jansen, S. Brinkkemper, J. Souer, and L. Luinenburg. Shades of gray: Opening up a software producing organization with the open software enterprise model. *Journal of Systems and Software*, 85(7):1495–1510, 2012.
[42] S. Jansen and M. Cusumano. Defining software ecosystems: A survey of software platforms and business network governance. *4th International Workshop on Software Ecosystems (IWSECO)*, page 41, 2013.
[43] S. Jansen, A. Finkelstein, and S. Brinkkemper. A sense of community: A research agenda for software ecosystems. In *31st International Conference on Software Engineering-Companion Volume, 2009.*, pages 187–190. IEEE, 2009.
[44] K. Kim and J. Altmann. Platform provider roles in innovation in software service ecosystems. *IEEE Transactions on Engineering Management*, pages 1–10, 2020.
[45] H.-B. Kittlaus and S. A. Fricker. *Software product management*. Springer, 2017.
[46] T. Lima, R. P. dos Santos, J. Oliveira, and C. Werner. The importance of socio-technical resources for software ecosystems management. *Journal of Innovation in Digital Ecosystems*, 3(2):98–113, 2016.
[47] K. Manikas and K. M. Hansen. Reviewing the health of software ecosystems–a conceptual framework proposal. In *Proceedings of the 5th International Workshop on Software Ecosystems*, pages 33–44. Citeseer, 2013.
[48] G. Müller-Seitz. Leadership in interorganizational networks: a literature review and suggestions for future research. *International Journal of Management Reviews*, 14(4):428–443, 2012.
[49] R. L. Paquin and J. Howard-Grenville. Blind dates and arranged marriages: Longitudinal processes of network orchestration. *Organization Studies*, 34(11):1623–1653, 2013.
[50] K. Petersen, S. Vakkalanka, and L. Kuzniarz. Guidelines for conducting systematic mapping studies in software engineering: An update. *Information and Software Technology*, 64:1–18, 2015.
[51] K. Popp and R. Meyer. *Profit from software ecosystems: Business models, ecosystems and partnerships in the software*

*industry*. Books on Demand, 2010.
[52] K. M. Popp. Goals of software vendors for partner ecosystems–a practitioner's view. In *International Conference of Software Business*, pages 181–186. Springer, 2010.
[53] P. Ralph and R. Robbes. The acm sigsoft paper and peer review quality initiative: Status report. *ACM SIGSOFT Software Engineering Notes*, 45(2):17–18, 2020.
[54] D. Riehle. The economic motivation of open source software: Stakeholder perspectives. *Computer*, 40(4):25–32, 2007.
[55] P. Ritala and A. Almpanopoulou. In defense of 'eco'in innovation ecosystem. *Technovation*, 60:39–42, 2017.
[56] P. Ritala, L. Armila, and K. Blomqvist. Innovation orchestration capability—defining the organizational and individual level determinants. *International Journal of Innovation Management*, 13(04):569–591, 2009.
[57] R. W. Scholz and O. Tietje. *Embedded case study methods: Integrating quantitative and qualitative knowledge*. Sage, 2002.
[58] Z. Spijkerman and S. Jansen. The open source software business model blueprint: A comparative analysis of 10 open source companies. In *SiBW*, pages 128–143, 2018.
[59] G. Tayauova. Advantages and disadvantages of outsourcing: analysis of outsourcing practices of kazakhstan banks. *Procedia-Social and Behavioral Sciences*, 41:188–195, 2012.
[60] A. Tiwana. *Platform ecosystems: Aligning architecture, governance, and strategy*. Newnes, 2013.
[61] A. Tiwana. Evolutionary competition in platform ecosystems. *Information Systems Research*, 26(2):266–281, 2015.
[62] A. Tiwana. Platform synergy: Architectural origins and competitive consequences. *Information Systems Research*, 29(4):829–848, 2018.
[63] G. Valença, C. Alves, and S. Jansen. A power perspective on software ecosystem partnerships. In *International Conference on Product-Focused Software Process Improvement*, pages 69–85. Springer, 2016.
[64] J. Van Angeren, C. Alves, and S. Jansen. Can we ask you to collaborate? analyzing app developer relationships in commercial platform ecosystems. *Journal of Systems and Software*, 113:430–445, 2016.
[65] J. Van Angeren, J. Kabbedijk, S. Jansen, and K. M. Popp. Partnership characteristics within large software ecosystems. In *Proceedings of the Third International Workshop on Software Ecosystems*, volume 37, 2011.
[66] J. Van Angeren, J. Kabbedijk, S. Jansen, and K. M. Popp. A survey of associate models used within large software ecosystems. In *Third International Workshop on Software Ecosystems*, pages 27–39, 2011.
[67] P. van Vulpen. *SECO-PMF Adaptation*. Mendeley Data, 2021, `http://dx.doi.org/10.17632/f7bx5vpy7t.1` [Dataset].
[68] P. van Vulpen. *Partner management Framework, key concept identification, clustering, merging and removal*. Mendeley Data, 2021, `http://dx.doi.org/10.17632/hm25k9sdm4.1` [Dataset].
[69] P. van Vulpen. *Partner Management Query Results including inclusion/exclusion reason*. Mendeley Data, 2021, `http://dx.doi.org/10.17632/t57pws28rp.1` [Dataset].
[70] P. van Vulpen, S. Brinkkemper, S. Jansen, and G. Lucassen. Continuous software portfolio performance management. In *International Conference of Software Business*, pages 75–89. Springer, 2018.
[71] P. Van Vulpen, A. Menkveld, and S. Jansen. Health measurement of data-scarce software ecosystems: A case study of apple's researchkit. In *International Conference of Software Business*, pages 131–145. Springer, 2017.
[72] M. Viljainen and M. Kauppinen. Software ecosystems: A set of management practices for platform integrators in the telecom industry. In *International Conference of Software Business*, pages 32–43. Springer, 2011.
[73] T. Weiblen and H. W. Chesbrough. Engaging with startups to enhance corporate innovation. *California Management Review*, 57(2):66–90, 2015.
[74] D. Werth, A. Emrich, and A. Chapko. An ecosystem for user-generated mobile services. *Journal of Convergence*, 3(4):35–40, 2012.
[75] J. West and D. Wood. Creating and evolving an open innovation ecosystem: Lessons from symbian ltd. *Available at SSRN 1532926*, 2008.
[76] J. West and D. Wood. Evolving an open ecosystem: The rise and fall of the symbian platform. In *Collaboration and competition in business ecosystems*, pages 27–67. Emerald Group Publishing Limited, 2014.
[77] C. Wohlin. Guidelines for snowballing in systematic literature studies and a replication in software engineering. In *Proceedings of the 18th International Conference on Evaluation and Assessment in Software Engineering*, page 38. Citeseer, 2014.
[78] J. Wouters, J. Ritmeester, A. Carlsen, S. Jansen, and K. Wnuk. A seco meta-model. In *International Conference on Software Business*, pages 31–45. Springer, 2019.
[79] P.-L. Yin, J. P. Davis, and Y. Muzyrya. Entrepreneurial innovation: Killer apps in the iphone ecosystem. *American Economic Review*, 104(5):255–59, 2014.
[80] R. K. Yin. *Applications of case study research*. Sage, 2011.
[81] Y. Yoo, O. Henfridsson, and K. Lyytinen. The new organizing logic of digital innovation: an agenda for information systems research. *Information systems research*, 21(4):724–735, 2010.

## Appendix: Partner Management Activities Description and Indicators

| | Activity | Description | Indicators (The orchestrator does:) |
|---|---|---|---|
| A1 | **Create revenue model** | The orchestrator should decide how revenue that flows through the ecosystem can be organized in an efficient manner. Customers adopt the platform and extensions and provide revenue for the ecosystem. This revenue flows to the orchestrator and partners. The orchestrator should decide how the revenue is distributed among actors. Customers can be billed separately for use of the platform and use of the extension, or orchestrators do invoicing for partners and give the partners their share. Either way, the orchestrator should make explicit how revenue flows in the ecosystem. Furthermore, if the orchestrator is thriving, it may start offering loans to start-ups in the ecosystem, to encourage innovation and growth. | 1. Structure revenue streams to customers<br>2. Structure revenue streams to partners<br>3. Induce a revenue fee<br>4. Create an opportunity for partners to loan within the ecosystem |
| A2 | **Enforce revenue streams** | After the orchestrator has created a revenue model for the ecosystem, the next step is to have partners align with that revenue model. The orchestrator should centrally organize the revenue flows, and set contracts for the revenue flow. | 1. Force partners to align with the revenue model<br>2. Create contracts with partners on revenue streams |
| A3 | **Create partner certificates** | Partners that possess a certain skill, completed a certain training, or are part of a certain hub can show this to other partners and customers. The implementation of partner certificates is an opportunity for the orchestrator to increase the reputation of partners and send a feeling of appreciation to partners. | 1. Create Partner certificates<br>2. Set criteria for partner certificates<br>3. Set multiple different certificates depending on the partner expertise |
| A4 | **Certify partner applications** | The orchestrator certifies partner applications. The orchestrator does so by defining criteria and evaluating whether an extension developed by a partner meets these criteria. Certified applications benefit from the reputation of the orchestrator. | 1. Create application certificates<br>2. Set criteria for application certificates<br>3. Set multiple certificates depending on the nature of the extension |
| A5 | **Create ecosystem principles** | Because an ecosystem bases itself in the collaboration of multiple actors, actor interaction is essential. The orchestrator sets standards of behavior, in collaboration with partners. These standards of behavior define the amount of interaction between partners, competition versus cooperation, attitude and the general tone in the ecosystem. | 1. Set ecosystem principles |
| A6 | **Avoid orchestrator incursion by contracts** | If the platform is further developed, it may take over functionalities that were originally provided by an extension from a partner. This pushes the partner out of its own market, and provides the orchestrator with the niche that was originally from the partner. Although this seems like an opportunity to diversify the platform and reach new customers, the orchestrator should take care as orchestrator incursion is destructive for the trust of partners. Therefore, the orchestrator avoids orchestrator incursion by contracts. If it is needed for the future of the platform that a certain partner extension is added to the core platform, that partner is informed timely of this. | 1. Discuss platform strategy with partners<br>2. Identify partners that produce potentially redundant extensions<br>3. Sign contracts regarding orchestrator incursion with partners<br>4. Contact affected partners and discuss how the platform and the extension can complement each other |
| A7 | **Set up personal partner contact** | The orchestrator maintains personal contact with the employees of partners. Although personal contact in itself does not lead to benefits in the ecosystem and does not scale easily, personal contact makes all kinds of interactions easier, and creates trust in the orchestrator. | 1. Maintain personal contact with partners<br>2. Designate a single partner manager to a company |
| A8 | **Develop ecosystem communication channels** | In order to effectively communicate with partners and consumers, the orchestrator creates ecosystem communication channels. Some ecosystems require personal contact with partners and customers. Other ecosystems require no personal contact, or formal interactions. The orchestrator defines how actors interact in the software ecosystem, and oversees the effects of the chosen communication channel. | 1. Develop communication strategy with partners<br>2. Enforce ecosystem communication strategy |
| A9 | **Create new partnerships** | To improve the sustainability of the ecosystem, the orchestrator identifies which actors are relevant for growing the ecosystem. These are actors that contribute to the robustness of the ecosystem or because their technology is of vital interest in the future. The orchestrator attracts these partners to the ecosystem. | 1. Identify strategic niches<br>2. Identify strategic partners that fit in these niches<br>3. Approach potential partners<br>4. Attract strategic partners into the ecosystem |
| A10 | **Discuss ecosystem participation value with partners** | The orchestrator reveals to partners and potential partners what participating in the ecosystem brings the partner. This is compared to not participating and to participation in rivalling ecosystems. The orchestrator should pinpoint the basic functionalities of the platform, but also reveal what the ecosystem can be used for in broader sense. | 1. Share platform capabilities with partners<br>2. Share ecosystem participation value with partners<br>3. Share ecosystem participation value with potential partners<br>4. Inform partners of the presence of other actors in the ecosystem |
| A11 | **Support non-technical partners in accessing the ecosystem** | Some partners may not have sufficient technical knowledge to develop an application on the platform themselves. When this is the case, the orchestrator steps in to ensure that the domain knowledge of this partner still reaches customers. The orchestrator can do so by assisting the partner in development of an extension. | 1. Set up an assistance program<br>2. Reach out to partners in need of assistance |
| A12 | **Create a community forum** | The orchestrator creates a community forum. Partners and customers discuss how the platform and its extensions are used on this forum. Customers share their product desires, found bugs and other experiences. Partners get in touch with their customers, and do marketing through the forum. This enhances the participation of partners in the ecosystem. | 1. Provide a community forum<br>2. Allow partners to access this forum<br>3. Allow customers to access this forum |
| A13 | **Create partner hubs** | The orchestrator creates partner hubs by bringing partners with similar expertise together. Partner hubs are sub-communities where partners that have a shared expertise can discuss the platform. By doing so, a partner hub becomes an authority of a domain of the platform. The partner hub discusses the connection to the platform, market trends, technology and more. | 1. Identify partner niches with a shared expertise<br>2. Create partner hubs<br>3. Share responsibility with the hubs |
| A14 | **Organize conferences** | Conferences act as a means to share knowledge and create personal relationships between actors. Therefore, the orchestrator organizes conferences and meetups with actors, to make partners and customers aware of each other and of upcoming technologies. This increases partner-partner and partner-consumer interaction. | 1. Organize conferences for partners and consumers |
| A15 | **Train partners** | Not every partner has all skills required to be an effective complementor to the ecosystem. It is the duty of the orchestrator to nurture these partners to an effective complementor. Partners can be trained in multiple ways, as different skills are required to effectively participate in the ecosystem. The orchestrator can provide sales training, marketing training, development training and consultancy training. | 1. Set up training programs<br>2. Set up sales training<br>3. Set up marketing training<br>4. Set up development training<br>5. Set up consultancy training<br>6. Set up an independent training organization in the ecosystem |
| A16 | **Create shared R&D teams** | The orchestrator creates R&D teams that are independent of partners whose sole purpose is to increase the knowledge that is available in the ecosystem. These R&D teams help partners in creating knowledge, and make created knowledge freely available in the ecosystem. | 1. Set up ecosystem R&D<br>2. Create specified R &D teams<br>3. Establish mechanisms to enhance knowledge capture by partners |
| A17 | **Solve conflicts between partners** | Partners in a software ecosystem that develop overlapping extensions are in an coopetition relationship. This leads to tensions that, left untreated, can ruin the relationships and community in the ecosystem. The orchestrator solves these conflicts by establishing rules and formal procedures. | 1. Solve arising conflicts between partners<br>2. Set formal rules and procedures that aid in conflict solving |

| | Activity | Description | Indicators (The orchestrator does:) |
|---|---|---|---|
| A18 | **Point out excelling partners** | The orchestrator uses its reputation to point out excelling partners. This improves the partner's reputation and increases brand recognition for the partner. The orchestrator can decide which partners are marketed through a marketing fee, by selecting strategic partners, or by making marketing part of a partnership program. | 1. Point out partners in blogs<br>2. Create an application spotlight in the app store<br>3. Market partners in a newsletter<br>4. Create partner awards |
| A19 | **Share sales knowledge and channels** | The orchestrator has full authority over the sales channels that a platform has. If a partner meets certain criteria or pays a fee, it should get access to the sales channel. Furthermore, the orchestrator can share its knowledge of sales with partners that have less experience with this. This will increase the revenue of partners in the ecosystem. | 1. Identify partners with limited sales knowledge<br>2. Share sales knowledge<br>3. Set participation criteria<br>4. Open sales channels for partners that meet the participation criteria |
| A20 | **Share marketing channels** | The orchestrator helps partners in some aspects of doing business, such as marketing. By sharing the marketing channels that an orchestrator uses to promote its platform with partners, the orchestrator assists partners that do not have resources for creating independent marketing channels. | 1. Create an ecosystem website<br>2. Identify marketing channels<br>3. Identify partners eligible for sharing the marketing channels<br>4. Share marketing channels with partners |
| A21 | **Open platform to diverse markets** | The orchestrator actively looks to find new niches where the platform and the extensions can be sold. This allows the existing partners to sell to new customers, while also attracting partners with domain knowledge of the new niche. | 1. Identify boundaries of existing market<br>2. Identify potential niches for platform expansion<br>3. Open the platform to other niches |
| A22 | **Create joint marketing agreements in the ecosystem** | The orchestrator supports in finding new opportunities to market products by encouraging product collaboration through creating joint marketing agreements between product. Two partners that collaborate, or a partner-orchestrator collaboration tightens the bonds in the ecosystem. | 1. Facilitate Joint marketing agreements |
| A23 | **Create an app store** | The orchestrator creates an app store where partners advertise their extensions. Partners that meet criteria have their applications accepted. The orchestrator removes extensions from the app store that fail to meet these criteria. The extensions can be reviewed and rated by customers so performance of applications can be evaluated. | 1. Create an app store<br>2. Establish app store criteria<br>3. Implement ratings and reviews by customers |
| A24 | **Recommend partners to customers** | In a software ecosystem where product configurations may reach endless complexity, customers find themselves asking which product extensions satisfy their needs. The orchestrator helps both customers and partners in this process by actively redirecting customers to partners, and revealing which extensions are available to customers. | 1. Enable customers to search for information of partners and extensions<br>2. Establish a customer helpdesk for selecting extensions<br>3. Enable customer redirection as an automatic process where the customer is pointed at partners automatically |
| A25 | **Bundle third-party applications and platform** | The orchestrator offers complete solutions to customers by bundling the platform with partner extensions. Partners are helped by the orchestrator in selling their product. This tightens the bonds and improves communication in the ecosystem. | 1. Bundle extensions in packages<br>2. Sell the platform as an integrated system |
| A26 | **Create public recognition for the ecosystem** | The orchestrator does large acquisitions or reaches media attention in other ways and becomes known by customers. Partners that co-develop with an orchestrator that reaches media attention will become more reputable themselves. | 1. Seek media attention |
| A27 | **Share reputation with partners** | The orchestrator has created a reputation as a larger company with a platform where partners can build extensions. The ecosystem benefits from this brand name when partners use this brand name for their own products. Partners that develop under the brand of a reputable orchestrator will become more reputable themselves. | 1. Share brand name with partner<br>2. Share brand name with third-party applications<br>3. Create a branding fee |
| A28 | **Create ecosystem roadmap** | The orchestrator acts as a visionary leader and reveals where the platform is headed. This enables partners to adapt to this platform and have certainty over the next time period. | 1. Create a roadmap<br>2. Collaborate with partners to create a shared vision<br>3. Share the roadmap with partners |
| A29 | **Force Partners to Align strategies with roadmap** | In order to ensure that partners will capture sufficient value from the ecosystem, partners should align their product strategies with the ecosystem roadmap. Otherwise, they will find themselves at risk of failing to meet market needs, or developing redundant extensions. Therefore, the orchestrator enforces partners to align with the ecosystem roadmap. By aligning the strategies of all partners with the roadmap, the ecosystem becomes coherent. | 1. Communicate with individual partners how they align with the roadmap<br>2. Fine partners that fail to align with the roadmap |
| A30 | **Discuss future platform technologies with partners** | The orchestrator oversees the innovations and technology trends that will impact the ecosystem. By discussing these with the partners that are affected by the trends, the partners can prepare themselves. Furthermore, this increases the cohesion of the ecosystem. | 1. Create a technical roadmap<br>2. Share technical trends with affected partners |
| A31 | **Expand technical possibilities of the platform** | The orchestrator should increase the means of how customers, partners and the orchestrator can interact with the platform. This allows the platform to be sold to new customers, allows new partnerships, and the orchestrator can set a higher license fee for customers | 1. Identify opportunities for technical extension<br>2. Develop technical extensions<br>3. Assign partners for the identification or development of platform extensions |
| A32 | **Create technology that assists in application development** | The orchestrator has created tools and technologies that facilitate the development of platform extensions. The orchestrator can also attract partners that assist in application development. Partners benefit from this by a faster development and a shorter time to market of ideas. | 1. Provide ecosystem specific tools for development of extensions<br>2. Share development tools with new partners<br>3. Create sandbox environment<br>4. Create developer forum<br>5. Attract partners that assist in application development |
| A33 | **Outsource core ecosystem functionalities** | In a further attempt to outsource development duties and create more time to orchestrate, the orchestrator outsources parts of the ecosystem to partners. The orchestrator is no longer responsible for development of part of the platform. This creates trust in partners, divides responsibility, and allows the orchestrator to orchestrate. | 1. Attract partners that take over platform development<br>2. Attract actors that take over training<br>3. Establish damage control and quality check measures |
| A34 | **Assist partners in platform updates** | The orchestrator communicates to partners whenever they need to adapt to a new platform version. This notification comes early enough for partners to prepare for a new version of the platform. | 1. Find affected partners by platform updates<br>2. Inform partners of new platform updates in time |
| A35 | **Create a centralized repository of knowledge** | the orchestrator has set up a centralized repository of knowledge. Doing so creates opportunities to share knowledge. Partners no longer have to find knowledge themselves. Furthermore, Partners that realize a certain type of knowledge is also beneficial for other partners can add this to the repository. The orchestrator can discuss trends and opportunities with partners. External repositories such as StackOverflow can act as a centralized knowledge repository. | 1. Create a knowledge repository<br>2. Encourage partners to share knowledge in the repository |

| | Activity | Description | Indicators (The orchestrator does:) |
|---|---|---|---|
| A36 | **Share knowledge required for ecosystem participation** | Decreasing the time for potential partners to gain access to the platform is beneficial as this reduces switching costs. By sharing integration knowledge through guides or tutorials, partners spend fewer resources on gaining access to the ecosystem. | 1. Share ecosystem participation knowledge |
| A37 | **Encourage knowledge sharing** | The orchestrator sets formal mechanisms in place to enforce knowledge sharing between partners. Knowledge regarding customer groups, marketing, or sales aids all partners. When partners collaborate in knowledge sharing, the collaborations in the ecosystem increase. | 1. Encourage partners to share knowledge |
| A38 | **Build industry consortium** | After the orchestrator has established a centralized repository, creating an industry consortium is the next step in gathering knowledge and credibility. The consortium consists of representatives of all stakeholders in a software ecosystem and its sphere of influence. This consortium discusses the future of the ecosystem, along with future technologies and solutions. The consortium brings increases innovation in the ecosystem, by pooling resources and knowledge. | 1. Identify consortium members<br>2. Invite consortium members<br>3. Create an ecosystem future vision with the consortium<br>4. Create a technical roadmap with the consortium |
| A39 | **Define entry requirements** | The ecosystem consists of a variety of actors. To ensure that actors adhere to a minimum quality level, the orchestrator should set entry requirements. After meeting these requirements an actor may join the ecosystem and start collaborating with other actors and sell to customers. One of the requirements can be a participation fee for partners that access the ecosystem. | 1. Set participation quality requirements<br>2. Set a participation fee |
| A40 | **Stimulate Complementary Partner Investments** | Partners should be discouraged to participate minimally in the software ecosystem. Instead, the orchestrator rewards partners that do complementary partner investments, are active in the forums, or participate at conferences. | 1. Create a reward for prolonged ecosystem participation |
| A41 | **Differentiate Partners** | The orchestrator can sets up multiple partner categories to adapt to a partner's specific wishes and needs. The partner can choose which category it belongs to, or the orchestrator can choose which category the partner belongs to. The partner should receive benefits according to the partner category it belongs to. | 1. Create partner categories<br>2. Adapt support per category |
| A42 | **Create partnership programs** | To tighten the bonds between the orchestrator and the partners, the orchestrator sets up partnership programs. The partner will receive knowledge, marketing or funding from the orchestrator in return for commitment to the platform. | 1. Create partnerships<br>2. Set rewards for partnerships<br>3. Set rewards for loyalty |
| A43 | **Imitate competing platforms** | The orchestrator investigates which technologies a competing ecosystems uses for developing applications, interacting with partners and creating value for customers. The orchestrator reviews if any of these technologies are relevant for the its ecosystem and implement these technologies. | 1. Investigate rival ecosystems<br>2. Compare performance between ecosystems<br>3. Implement identified technologies |
| A44 | **Align with complementing ecosystems** | Aligning standards with upstream or downstream ecosystems reduces partner investments for vertical integration. The orchestrator should create a common development tool with a complementing ecosystem, to encourage multi-homing. By aligning with complementing ecosystems, partners may develop the entire supply chain to customers. | 1. Identify upstream and downstream ecosystems<br>2. Align standards with complementing ecosystems<br>3. Create a common development tool |
| A45 | **Promote adoption of key technologies and related standards** | The orchestrator ensures that partners adopt key technologies. This enables the ecosystem to be in line with market trends. Additionally, adopting technologies and standards throughout the entire ecosystem eases collaboration and interaction between actors. | 1. Identify key technologies<br>2. Encourage partners to adopt key technologies<br>3. Fine partners that do not adopt ecosystem standards |
| A46 | **Share innovations with partners** | It is the responsibility of the orchestrator that the ecosystem and its partners remain on top with technology trends and in line with market needs. The orchestrator develops new technology or attracts partners that possess this technology. The orchestrator then shares these innovations with partners. | 1. Facilitate the creation of new technology<br>2. Encourage partners to use the new technology |
| A47 | **Support the protection of Intellectual Property Rights** | Partners should see the ecosystem as a safe haven but may see the sharing of knowledge in the ecosystem as a threat to their own knowledge, especially when sharing to rivals. The orchestrator supports these partners in their knowledge sharing by establishing formal mechanisms that protect the IPR of partners. | 1. Define mechanisms to protect IPR |
| A48 | **Co-develop with partners** | Experimentation can lead to new business opportunities in the ecosystem. the orchestrator leads in this by setting up opportunities for experimentation in collaboration with partners. After the initial experiments have taken place, the collaborative agreement between orchestrators and partners sets out to conduct pilots. A pilot is an experiment that is applied to customers. This finds new niches in the markets that can be fulfilled by the software ecosystem. Finally, knowledge created in the software ecosystem should stay in the software ecosystem, and ideally, be shared freely between partners. The orchestrator enforces this by creating contracts that allow knowledge to be shared among partners. | 1. Identify partners to experiment with<br>2. Create experiments in collaboration<br>3. Conduct pilots with partners<br>4. Create IPR with partners |