

# Software Ecosystems: From Software Product Management to Software Platform Management

Slinger Jansen, Stef Peeters, and Sjaak Brinkkemper

Department of Information and Computing Sciences  
Utrecht University, the Netherlands  
{s.jansen, s.brinkkemper}@cs.uu.nl; stefpeeters86@gmail.com

**Abstract.** Presently, it is impossible to use software product management practices and tools for software platforms that operate in software ecosystems. The extensive and mature Software **Product** Management Competence Model cannot easily be applied in this context. In this paper the Software Product Management Competence Model is ported towards keystone players in software ecosystems, to create the new Software **Platform** Management Competence Model. If a keystone player implements the capabilities described in these new tools, it can manage stakeholders, know and align their interest, and thereby further enable value creation by itself and ecosystem members.

**Keywords:** Software Product Management; Software Ecosystems; Keystone; Partners; Directed Approach; Software Platform Management

## 1 Introduction

When software products grow larger, external parties may wish to extend products to create niche solutions for niche markets. The product grows to a platform on which third parties build extensions, components and applications. Iansiti and Levien (2004) define a platform as a set of standard services, tools, and/or technologies that function as resources for other members. In this case, the software company and its platform provide core technology that function as a basis for a Software Ecosystem (SECO). Jansen, Finkelstein and Brinkkemper (2009) have defined a SECO 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.*”

If a software producing organization is going to manage its product *as a platform* it is taking a directed SECO approach. Cusumano (2010) presents several reasons for software companies to take a directed SECO approach, such as when (potential) customers come from a plethora of niche markets. Due to time constraints and limited R&D investment budget, satisfying different types of customers with software that fit their needs cannot be solely done by one keystone software provider. Expanding the product to a platform on which externally created components or applications can be built is an instrument for creating the required niche functionality and is a basis for the creation of a SECO.

Software Product Management (SPM) plays a key role in product software organizations in creating successful products. Organizing SPM well increases the quality of the developed products. Researchers have instantiated many initiatives to improve the activities with regard to SPM. The results of these studies have led to an approach to organize the management of requirements, releases, products and portfolio of products; i.e. a complete framework with all relevant management practices (van de Weerd, Brinkkemper, Nieuwenhuis, Versendaal, & Bijlsma, 2006).

Taking a SECO approach affects how companies look at their SPM. Those software companies that are able to manage their software platform, relationships and surrounding environment in a successful manner create a sustainable and profitable business for themselves and their stakeholders. The current SPM framework already contains practices that take into account external parties, such as “*Monitor Partner Network*”. The current model, however, does not sufficiently accommodate companies that take a directed SECO approach, as it lacks, for instance, the identification of SECO player types, the use of multiple distribution channels (app stores), and the certification of partners to develop for and on top of the platform.

According to Bosch (2009) the keystone (i.e. platform developer) can take a SECO approach ranging from directed to undirected. If a keystone takes a directed approach it identifies specialized market segments (i.e. niche markets) to offer solutions for and collaborates with third parties to develop domain specific functionality. In the undirected approach every external party that wishes to build new functionality on or with the offered platform can do so without permission of the keystone. This study focuses on SPM for software producing organizations with a directed SECO approach. To determine how SPM with a directed SECO approach (i.e., Software Platform Management) needs to be organized, we investigated the adequacy of the instruments of the state of art of SPM, and if it is not, what needs to be changed. The instruments are:

- The SPM Competence (SPMC) Model: a framework which presents an overview of all important areas and practices for SPM (Bekkers, van de Weerd, Spruit, & Brinkkemper, 2010; van de Weerd et al., 2006).
- The SPM Maturity (SPMM) Matrix (Bekkers & van de Weerd, 2010): a maturity matrix in which all practices of the SPMC Model are presented in a best practice order for implementation. The SPMM is not presented in this paper for reasons of brevity.

We have found several clues that suggest that the model and matrix are not adequate for application in a company that is a keystone player. First, partner requirements in SECOs may be of a higher priority than customer requirements. While a customer only represents the value of one customer, partners usually have many customers and thus represent a larger possible value for the keystone organization. Second, it may affect release planning because the configurations of features in new releases have consequences for the externally created components. To synchronize releases with partners, the creation of widely accepted release definitions is essential. Third, besides a roadmap for the platform of the keystone organization each partner may have a roadmap for its solution(s). If the roadmaps in the SECO are not aligned correctly, it can lead to major problems. For example, the vision for the platform

might not be reconcilable with solutions created by partners. On the other hand, the platform company does not want to get in a ‘release stasis’ as well, in which it does not dare to innovate its platform. Fourth, the externally built solutions may compete with other products built by the organization. These problems and many more may arise when taking a directed SECO approach, which are explored.

The paper continues with a description of how design science was used for the research design. Furthermore, interviews and a questionnaire were used to find legitimate candidate changes for the tools and models. In section 3 and 4 related literature on the topics of SPM, the SPMC Model, the SPMM Matrix and SECOs is described. In section 5 the results of the study are presented, with an emphasis on the SPMC Model. Section 6 discusses the results and shows the weaknesses in the research approach in terms of validity. In Section 7 is concluded that the SPLaM is future proof and enables keystone players in software ecosystems to better manage their platform within the complete ecosystem. Furthermore, future research topics are suggested that further develop the maturity models in the domain of software platform management.

## 2 Research Approach

For this study is chosen to use the Design Science Research Cycle as described by Takeda, Veerkamp, Tomiyama, and Yoshikawa (1990). Five steps are defined that represent the design research process. The five steps are: 1) awareness of the problem, 2) suggestion, 3) development, 4) evaluation and 5) conclusion.

First, during the step ‘Awareness’ a relevant problem to conduct a study was found. Second, during the step ‘Suggestion’ was suggested what solutions could possibly solve the stated problems. A literature study on the topics of the SPMC Model, the SPMM Matrix, SPM, SECOs, and activities related to these research domains was performed.

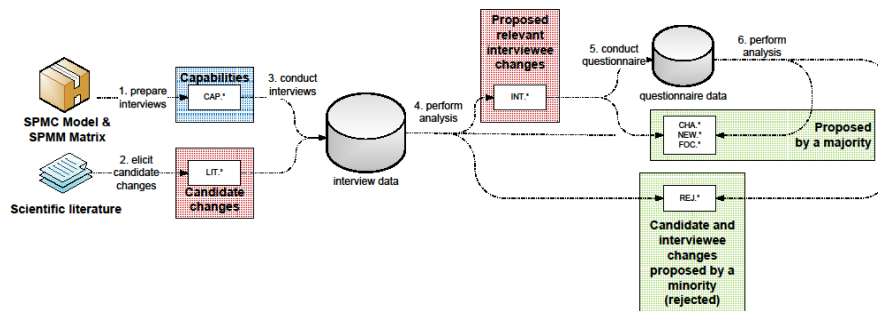


Fig. 1. Research steps in the development of the SPLaM.

The literature study was used as the knowledge base on which eleven structured interviews with Dutch product managers were conducted. The interview data were classified per topic within the SPM and SECOs literature, as to collect as many facts about specific SPM practices. The interviews consisted of an introduction, some generic questions about software platforms and platform management, a deep-dive into the SPM model, and a wrap-up. During the discussions surrounding the SPM model, proposed changes were documented carefully. Interviews took an average of 2 hours, mostly because of the lengthy discussions about the SPM model. Interviews were typically interrupted by a coffee break.

**Table 1.** Interviewees and experience.

#	Experience in				Gained at
	RM	RP	PP	PM	
<b>i1</b>	x	x	x		Ordina, Infra Design, Unisys
<b>i2</b>	x	x	x	x	PinkRocade
<b>i3</b>	x	x	x	x	Everest
<b>i4</b>	x	x			NetAspect
<b>i5</b>	x	x	x	x	ThinkWise
<b>i6</b>	x	x			AFAS Personal, Yunoo
<b>i7</b>	x	x	x	x	Everest
<b>i8</b>			x	x	ANVA, Cap Gemini
<b>i9</b>	x	x	x	x	BackBase, SDL Tridion, Pallas Athena, Data Distilleries
<b>i10</b>	x	x	x	x	Everest
<b>i11</b>			x	x	Everest
<b>Sum</b>	<b>9</b>	<b>9</b>	<b>9</b>	<b>8</b>	

Third, during the step ‘development’ the data of the interviews was used to create an overview of all proposed changes from literature and from the interview data. This resulted in several pages of proposed changes, which were used to create a draft version of the SPlaM. In those cases where proposed changes were conflicting, the interviewees were contacted to iron out differences. Practically all definitive changes in the resulting model were made based on what the majority of the product managers wanted to see changed. However, some minor and relevant changes (e.g. expanding the examples in the description of a SPM activity) were performed without a majority vote from the software product managers.

Fourth, during the step ‘Evaluation’, the previous interviewees were asked to evaluate the new model in a pen-and-paper survey where the researcher was present. Each of the proposed changes was evaluated. Furthermore, the main evaluation questions were:

- Which candidate changes are not relevant for a directed SECO approach?
- Which candidate changes still require additions or clarifications? Are they made in the right places of the SPM model?
- Which candidate changes are still missing in your opinion?

As the model had been discussed before with the interviewees, the evaluation did not lead to heated discussions. Finally, during ‘Conclusion’ step, the study is concluded by presenting the study results to the scientific community. An overview of the interviewees’ experiences in the different fields can be found in Table 1 and shows an even spread across the four disciplines in SPM.

### **3 Software Product Management**

The capacity of a software product to satisfy the needs and expectations of stakeholders determines its quality (Berander, 2007). Therefore Berander (2007) states, a software producing organization needs to gather, select and plan the right set of features for a product to find the highest value for all stakeholders. However, SPM covers more responsibilities. SPM concerns the definition of a strategy for, distribution of, launching of, providing support for, and phasing out of a product; i.e. all phases of the life-cycle of product software (Ebert, 2007). The same author says about SPM that it assures ‘winning’ products by implementing business cases, agreeing on and implementing marketing, creating functional and technical roadmaps, managing product life-cycles, and aligning and optimizing the organization’ product portfolio.

The SPM Competence Model (Bekkers, van de Weerd, Spruit, & Brinkkemper, 2010; van de Weerd et al., 2006) gives an overview of the key areas of SPM. Its objective is threefold: aiding software companies in organizing and enhancing their SPM, structuring education on SPM and structuring research on SPM. The model consists of four business functions: Portfolio management, Product planning, Release planning and Requirements management. Its structure is chosen because a software producing organization possesses a portfolio of products, which consists of one or more products, which has multiple releases, and a release represent a set of requirements. Development activities are not part of the model, simply because it is not part of SPM. Each business function consists of a highly cohesive group of focus areas, which in turn represent a group of highly cohesive capabilities (i.e. relevant SPM practices). The external stakeholders are the: market, customers, and partners. The internal stakeholders are the (business units): company board, sales, marketing, research & innovation, development, support, and services.

The SPM Maturity (SPMM) Matrix (Bekkers, van de Weerd, Spruit, & Brinkkemper, 2010) is based on the SPM Competence Model; the matrix has the same structure (i.e. the business functions) and the same components (i.e. the focus areas and capabilities). However, the capabilities on which the focus areas are based are spread in a best practice order for implementation over several maturity levels. In this way, product managers and software organizations can determine how mature their SPM organization is (i.e. the level corresponding to their capabilities) and what the areas of improvement are (i.e. the missing capabilities corresponding to the desired level).

## 4 Software Ecosystems

The term Software Ecosystems (Jansen et al., 2013) and their underlying theory are based on biological ecosystems. The biological ecosystem is the result of the interactions between its members and the physical environment (Dhungana, Groher, Schludermann, & Biffel, 2010). SECOs involve the organization of its members (i.e. software vendors, third-party developers, suppliers and users) and its platform (J. Bosch, 2009).

Taking a directed SECO approach implies that the platform developer takes a community perspective, in which internal and external stakeholders are taken into account (Fricker, 2010). A well-known example of a successful ecosystem is Apple with its Appstore. The large quantity and quality of the product software (apps) offered in this store could not be devised and produced by Apple on its own. The success of this and other ecosystems lie in the opportunity for a large set of developers to use the platform to create and distribute software.

Two key types of members in SECO literature are recurring (van der Schuur, Jansen, & Brinkkemper, 2011); the keystone and the niche players. Iansiti and Levien (2004) state the keystone is: "...a benevolent hub in the network that provides benefits to the ecosystem and its members." They say it typically gives other members (i.e. niche players) the necessary space to grow and prosper and niche players (i.e. the other members of the network) do not try to compete with the keystone. They leverage the resources of the network to create solutions which are targeted at niche markets (i.e. specialized segments of the market). Thus, the keystone creates and delivers a keystone product (i.e. the platform) and surrounding services, which enable niche players to create and deliver niche solutions.

## 5 Software Platform Management

Many authors use the term platform when they talk about the keystone its product (and surrounding services) that is provided to enable other members to create value (e.g. in Geir K., 2011; Iansiti & Levien, 2004; Iyer, Lee, & Venkatraman, 2006; Kilamo, Hammouda, Mikkonen, & Aaltonen, 2012). The keystone opens its product to external entities to create a platform by which business and SECO objectives can be reached. Thus, management needs to be focus on how the keystone and other members of the SECO can create value. Kittlaus and Clough (2009) named this approach platform planning. Companies that conduct successful platform planning will realize several benefits (Robertson and Ulrich, 1998), they will:

- have a greater ability to create niche products for niche markets or customers;
- lower the costs to reach niche markets or customers;
- create niche products that more closely meet the needs of them.

We define *platform management* as consisting of four processes: Portfolio management, Product planning, Release planning, and Requirements management. Since

these platform planning processes are similar to those in the SPM framework, the framework under development is called the Software Platform Management (SPlaM) framework.

## 5.1 Candidate Changes

Based on the literature candidate changes for the SPMC Model are defined, classified in the following topic groups: foster the sharing of resources, manage the involvement of partners, manage the communication of requirements, raise the quality by means of certification, initiate and manage (new) partner relationships, create a healthy SECO product portfolio, and initiate and manage (new) SECO (sales and distribution) channels. A large number of candidate changes were identified. For example, based on literature on the topic of “foster the sharing of resources”, a candidate change is made: the *identification of core assets* process is expanded to include core assets created by third parties as well (see Table 2). Imagine, for instance, the role of the Facebook application on the iPhone, which can be considered a core asset in the iPhone ecosystem, even though it was not released and developed by Apple.

**Table 2.** Candidate change core asset identification.

focus area	candidate change	processed as
Core asset roadmapping	Common components/functionality (core assets) is systematically identified among the ecosystem’s products and deliverables surrounding these products.	Expand capability b. Core asset identification with this candidate change or add it as a new capability.

## 5.2 SPlaM: The New Model

As explained earlier the current model and matrix consists of four business functions: Requirements management, Release planning, Product planning and Portfolio management. No (new) business functions are added or removed. Although the high level business functions are not changed, new focus areas and capabilities are added and existing focus areas and capabilities are changed. The following tables (**Table** and **Table**) describe what changes were made to the model. In the ‘part’ column is described what is changed; it can be the name of a new or changed capability or the description of a focus area. In the ‘c/n’ column is described if it is a new (i.e. ‘n’) or changed (i.e. ‘c’) capability or focus area. For the sake of brevity **unchanged** capabilities, **unchanged** focus areas and (**changing**) maturity levels are not presented. For more information on the unchanged capabilities and focus areas, please see Bekkers and van de Weerd (2010). All changes made to (the description of the) focus areas are based on the changes made with regard to new and changed capabilities.

A final remark has to be made about the *Partnering & contracting* focus area of the current SPMC Model. Due to the fact that nine new capabilities are added to it, it is split up into three new focus areas. The three new focus areas are *Contracting*, *Partnering* and *Channel development*. Three of the five capabilities of the ‘old’ *Partnering & contracting* focus area are unchanged and therefore not described in the

following tables. The unchanged capability *Intellectual property management* is added to the *Contracting* focus area and the unchanged capabilities *Establish and evaluate pricing model* and *Investigate distribution channels* are added to the Channel development focus area. The interviewees mentioned several interesting observations while proposing changes to the SPMC model.

One specific aspect that was frequently discussed is partner trust: *“There exists a significant danger in opening up the requirements database to anyone.”* Furthermore, it was added that the type of access is relevant: *“Even if you intensely collaborate with partners, you don’t want them to be able to delete requirements from your database.”* Also, trust does not only play a role between the organization and its partners, but also partners among themselves: *“It’s utopian to think that partners will not compete amongst themselves. You can try all you want, there is always competition and you do not want to be an arbiter in an endless fight.”*

Another aspect mentioned frequently is transparency: *“Sometimes you don’t even know your customer base is interested in a specific feature until one proposes it and others get to comment on the feature. Sometimes the customers themselves don’t even know until they see the idea from another customer.”*

The most positively evaluated addition to the SPMC model is the certification of partners: *“It provides stakeholders with an unmatched transparency. Customers will know that this is a trusted party and that their extensions are at least ok’d by us.”* Certification is also seen as a marketing tool: *“Partners can go through several phases: from unknown to registered to certified to preferred. They can even use this for marketing purposes, every promotion is a sign that the partner is a little higher on the ladder to partnership.”*

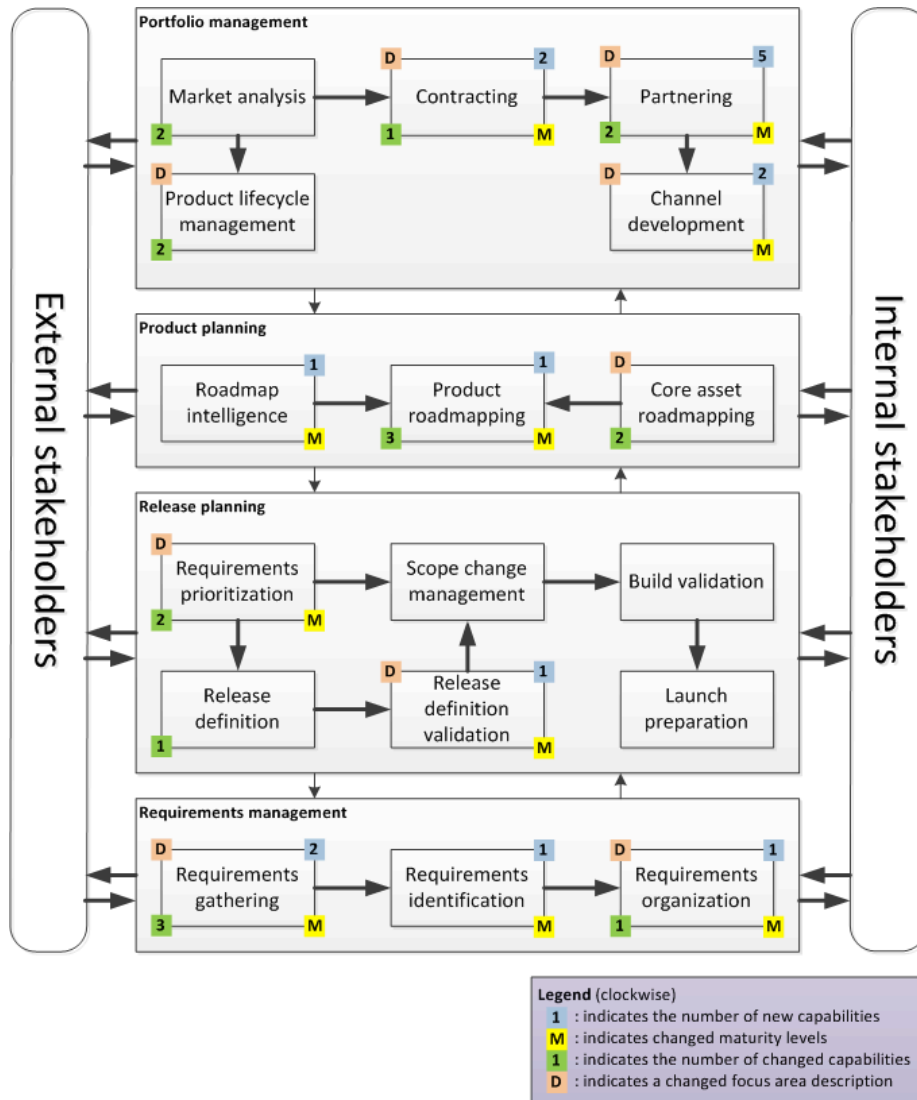


**Table 3.** Part one of the performed changes table.

<b>focus area</b>	<b>part</b>	<b>c/n</b>	<b>description</b>
<i>Requirements gathering</i>	[description]	c	Expanded with the sharing of requirements with relevant and authorized external stakeholders.
	<i>Opening central data-base</i>	n	The central database with incoming requirements is opened for relevant and authorized external stakeholders. It must foster the sharing of resources between SECO members.
	<i>Stakeholder involvement</i>	c	A combination of three pre-existing capabilities; i.e. the <i>Internal stakeholder involvement</i> , <i>Customer involvement</i> and <i>Partner involvement</i> capabilities. In it all relevant internal and external stakeholders are involved by gathering their requirements. Plus, per product or release is determined which stakeholder involvement is most important. Leading to the involvement of the right stakeholders.
	<i>Requirements communication flows</i>	n	The requirements communication networks are modeled and analyzed to determine the proper communication tactics for requirements.
<i>Requirements identification</i>	<i>External feedback</i>	n	Extra feedback on product requirements is gathered from external stakeholders. It raises the quality of product requirements by enriching its content.
<i>Requirements organization</i>	[description]	c	Expanded with sharing the gathered requirements with all relevant internal and external stakeholders.
	<i>Requirement organization</i>	c	Requirements are organized on shared aspects and requirements for externally build products are recognized and communicated to the specific external developer (i.e. partner). In this way, partners get all the relevant information they need to improve their niche solutions.
	<i>Opening requirements history log</i>	n	Make the history log accessible to relevant and authorized external stakeholders. Requirements will be reusable for other external projects and thereby fosters the sharing of resources within the SECO.
<i>Requirements prioritization</i>	[description]	c	The prioritization of requirements is only performed by relevant stakeholders.
	<i>Internal stakeholder involvement</i>	c	Still all relevant internal stakeholders indicate the requirements that should be incorporated in future releases. However, for each stakeholder is determined how important their involvement for the product is.
	<i>External stakeholder involvement</i>	c	A combination of two pre-existing capabilities; i.e. the <i>Customer involvement</i> and <i>Partner involvement</i> capabilities. In it, all relevant external stakeholders are involved by prioritizing the requirements and per product is determined which stakeholder its involvement is most important. Leading to the incorporation of the right requirements.
<i>Release definition</i>	<i>Communication</i>	c	The original <i>Internal communication</i> capability is expanded with external communication (i.e. communicating the release definition to external stakeholders as well). Now, partners will know what will be developed and they can prepare and/or improve their niche solutions based on the new or changed features.
<i>Release definition validation</i>	[description]	c	Expanded with external parties.
	<i>External validation</i>	n	The release definition is checked by external stakeholders as well. It creates a better alignment with externally created products, increases its quality, and generates awareness among the external stakeholders.
<i>Roadmap intelligence</i>	<i>Legislation</i>	n	Continuously an overview needs to be created with regard to changing legislation for the organization its product industry in order to keep compliant with laws and regulations.
<i>Core asset roadmapping</i>	[description]	c	Widened to the whole SECO.
	<i>SECO core asset identification</i>	c	The original <i>Core asset identification</i> capability is expanded to all products created in the SECO. Core assets are systematically identified among and surrounding the deliverables of SECO's products, because it increases and simplifies the reuse and maintenance of SECO its core assets.
	<i>Make, buy or co-creation decision</i>	c	The original <i>Make or buy decision</i> capability is expanded with co-creation decisions. A process needs to be in place to actively investigate make, buy or co-creation decisions, because costs can be reduced and time can be saved by using and/or co-create with external parties.
<i>Product roadmapping</i>	<i>Theme identification</i>	c	Release themes are identified and maintained together with relevant internal and external stakeholders for internal and external creation. It is expanded with external stakeholders, themes and creation, because external parties (i.e. partners) are going to develop the new value in a SECO.
	<i>Consultation</i>	c	The original <i>Internal consultation</i> capability is expanded to external stakeholders. Relevant internal and external stakeholders are consulted for the creation of a product roadmap. To have SECO wide acceptance of the product roadmap, to use the knowledge of all relevant members and to create richer and more realistic product roadmaps.
	<i>Long-term roadmap</i>	c	A long-term roadmap is created that spans a time period of maximum two years. The time span is shortened, because the software industry is changing so fast it is not possible to create valuable roadmaps that span more than two years.
	<i>Roadmap procedure</i>	n	A decision procedure has to be defined to make partners aware what will happen if no consensus is reach between the keystone and them in the future plans for the platform.

**Table 4.** Part two of the performed changes table.

<i>Market analysis</i>	<i>Market trend identification</i>	c	There is an active search for market opportunities to expand existing or create new products for, by doing market research at all kinds of places (e.g. related markets and visiting conferences). It is expanded by adding market research via the use of information gathered from partners, because in SECOs the keystone closely collaborates with the partners.
	<i>SECO customer win/loss analysis</i>	c	The original <i>Customer win/loss analysis</i> capability is expanded to all products in the ecosystem. A win/loss analysis is performed to determine why customers (of partners) did or did not choose to buy SECO products (i.e. the sales process is reviewed). To learn more about how to generate more customers by tuning the development of the platform.
<i>Contracting</i>	[description]	n	Focuses on establishing relations with external stakeholders by creating proper and clear agreements with them.
	<i>Service level agreements</i>	c	SLAs are set up for customers and partners. It is expanded to partners since they will ask for specific services on which agreement have to be made.
	<i>Contract negotiation process</i>	n	A contract negotiation process is set up in which (e.g.) realistic objectives, agreements on earnings, intellectual property rights, termination clauses, penalties for bad performance and arbitration procedure are determined.
	<i>Determine information profiles</i>	n	Information profiles are determined for each (type of) partner(s) (according to their role), it makes clear which partner has access to which information to simplify the sharing of information.
<i>Partnering</i>	[description]	n	Focuses on managing relations with external stakeholders and supporting them in creating the biggest possible value for the ecosystem.
	<i>Register partners</i>	n	All partners are registered in a central database which all (relevant) internal stakeholders can access, to create an overview of all partners and share knowledge (e.g. best practices and experiences) with regard to the partners.
	<i>Set up partner network</i>	c	The original <i>Monitored partner network</i> capability is split up in this capability and the new capability <i>Partner performance analysis</i> . Partner networks and/or portals are used to regulate and promote partnering.
	<i>Cluster partners</i>	n	Partners are clustered into groups with specific goals, functions, etcetera to simplify the management of them.
	<i>Coordinate partner alliances</i>	n	Partner(s) (alliances) are coordinated to avoid conflicts and to foster synergy to create a stronger and more coherent SECO.
	<i>Partner performance analysis</i>	c	The original <i>Monitored partner network</i> capability is split up in this capability and the new capability <i>Set up partner network</i> . A partner analysis is performed on an organizational level to analyze what partners have to offer, what their strengths and weaknesses are, and are going to offer. To create a clear and correct picture of the performance of partners which is the basis on which decisions can be made about maintaining or ending partner relations.
	<i>Certify partner</i>	n	Partners are certified divided over different ranks with different obligations and privileges to make clear what is expected to raise quality.
	<i>Certify external components</i>	n	Certify external created components on standard quality rules to raise the quality of niche solutions.
<i>Channel development</i>	[description]	n	Focuses on establishing and managing distribution channels.
	<i>Common delivery channel</i>	n	Set up a common delivery channel (e.g. the Apple Appstore) to enable partners to sell their products to a large customer base. It makes the SECO more attractive for new partners and customers.
	<i>Model the SECO</i>	n	Model the SECO at its different levels (e.g. within and between SECOs) to identify distribution channels, main competitors and potential partners.
<i>Product lifecycle management</i>	[description]	n	Widened to the entire SECO
	<i>SECO product lifecycle analysis</i>	c	The original <i>Product lifecycle analysis</i> capability is expanded to the whole ecosystem and by using information from external stakeholders as well. At least once per year the current life phase of each product in the SECO is determined based on technical and financial aspects. Plus, information is gathered from internal and external stakeholders. It is important to determine if the keystone still want to support the creation of certain niche solutions and therefore external stakeholders have to provide information on externally created products.
	<i>SECO portfolio scope analysis</i>	c	The scope of the original <i>Portfolio scope analysis</i> capability is widened to all products in the SECO. A product scope analysis is performed to identify overlaps and gaps between the products in the whole SECO, because it is important to create a healthy (i.e. diverse) SECO product portfolio.



**Fig. 2.** The Software Platform Management Competence (SPlaM) Model.

In **Fig. 2** the SPlaM Model is presented. The only changes that are visible in this figure are the three new focus area *Contracting*, *Partnering* and *Channel development*. The three focus areas are a substitution of the *Partnering & contracting* focus area in the old model. Per focus area is indicated if changes are made with regard to its description, new capabilities, changed capabilities and maturity levels. First, at the upper left corner is indicated in orange with the letter ‘D’ if its description is changed. Second, at the upper right corner is indicated in blue how many new capabilities are add to it. Third, at the lower left corner is indicated in green how many capabilities

are changed. Fourth, at the lower right corner is indicated in yellow if maturity levels of capabilities are changed.

## **6 Discussion**

To solve validity issues, tactics with regard to content validity were studying the instruments of the state of the art of SPM (i.e. the SPMC Model and SPMM Matrix), performing a literature study on the topics of SPM, SECOs, and activities related to these research domains, to now every relevant facet of the object under study. Tactics with regard to construct validity were using structured predefined approaches in which no room for own interpretations was possible during the gathering and analysis of the data. To ensure the product managers knew if and how they had to make changes, an introduction on the model, SECOs and the research objective was given with an introduction that was read to each interviewee. Tactics with regard to external validity are the use of multiple product managers from multiple companies as the developers and validators of the new model. A reduction of external validity results from the fact that only Dutch product managers are used as developers and validators. Thus, the question remains if the result of this study is generalizable to other countries and/or cultures. Fourth, the result is reliable because for every activity a predefined and structured approach is used. Thus, if this study were to be replicated, we expect it to lead to the same output.

During the interviews some of the product managers questioned whether there should be so much emphasis on the management of partner relationships within SPLaM. They indicated partnering activities are more appropriate at a partner management department. We have chosen to include these activities for the following reasons. First, other product managers did add new capabilities in regards to partnering activities. Furthermore, the current model already contains partner management capabilities. Thirdly, partners of keystone organizations with a directed SECO approach add value and knowledge to the ecosystem. Thus, a product manager needs to be deeply involved in partner management activities. Also, Fricker (2010) states: *“Software product management establishes and maintains a software ecosystem by managing stakeholders and studying and aligning their interests.”* Bekkers, van de Weerd, Spruit and Brinkkemper (2010) also indicate that the product manager is located at the center of the company; from its position she needs to keep contact with every relevant stakeholder to collaboratively reach goals derived from (business) strategy. Finally, Ebert (2007) describes that the product manager needs to find a balance between the needs and wishes of external entities (i.e. customer, markets and stakeholders) and guide them in the right direction.

## **7 Conclusion and future research**

The interviews resulted in fourteen new capabilities, sixteen changed capabilities, nine focus areas with changed maturity levels and nine focus areas with changed de-

scriptions. During the interview analysis, a selection was made of changes suggested by two or more product managers. These changes are presented to and assessed by the product managers by means of the questionnaire. The questionnaire resulted in two new capabilities, three changed capabilities and one focus area with changed maturity levels. The new, changed and unchanged capabilities and focus areas describe how keystone software organizations should organize their SPLaM practices with a directed approach. Thus, a Software Platform Management Competence (SPLaMC) is developed and validated.

The limitations of this study can be the initiation for further research. First, new studies could focus on what the relevant SPLaM practices are for software vendors that take an undirected SECO approach. Second, future research could focus on the matrix specific maturity levels and prerequisites of capabilities of the SPLaMM Matrix.

## References

- Bekkers, W., & van de Weerd, I. (2010). *SPM maturity matrix* (UU-CS-2009-015). Utrecht: Utrecht University.
- Bekkers, W., van de Weerd, I., Spruit, M., & Brinkkemper, S. (2010). A framework for process improvement in software product management. In A. Riel, R. O'Connor, S. Tichkiewitch & R. Messnarz (Eds.), *Systems, Software and Services Process Improvement* (pp. 1-12). Berlin Heidelberg: Springer.
- Berander, P. (2007). *Evolving prioritization for software product management*. Blekinge: Blekinge Institute of Technology Doctoral Dissertation Series.
- Bosch, J. (2009). From software product lines to software ecosystems. *Proceedings of the 13th International Software Product Line Conference*, San Francisco, CA, USA, 111-119.
- Cusumano, M. A. (2010). *Staying Power: Six Enduring Principles for Managing Strategy and Innovation in an Uncertain World*. Oxford University Press, USA
- Dhungana, D., Groher, I., Schludermann, E., & Biffel, S. (2010). Software ecosystems vs. natural ecosystems: Learning from the ingenious mind of nature. *Proceedings of the 4th European Conference on Software Architecture*, Copenhagen, Denmark. 96-102.
- Ebert, C. (2007). The impacts of software product management. *Journal of Systems and Software*, 80(6), 850-861.
- Fricke, S. (2010). Requirements value chains: Stakeholder management and requirements engineering in software ecosystems. In R. Wieringa, & A. Persson (Eds.), *Requirements Engineering: Foundation for Software Quality* (pp. 60-66). Berlin Heidelberg: Springer.
- Geir K., H. (2011). A longitudinal case study of an emerging software ecosystem: Implications for practice and theory. *Journal of Systems and Software*, 85(7), 1455-1466.
- Iansiti, M., & Levien, R. (2004a). *The keystone advantage*. Boston: Harvard Business School Press.
- Iyer, B., Lee, C. H., & Venkatraman, N. (2006). Managing in a small world ecosystem: Some lessons from the software sector. *California Management Review*, 48(3), 28-56.
- Jansen, S., Finkelstein, A., & Brinkkemper, S. (2009). Business network management survival strategy: A tale of two software ecosystems. *Proceedings of the 1st Workshop on Software Ecosystems*, Falls Church, Virginia, USA, 34-48.
- Jansen, S., Cusumano, M., Brinkkemper, S. (2013) *Software Ecosystems: Analyzing and Managing Business Networks in the Software Industry*. Edward Elgar Publishers, 350 pages.

- Kilamo, T., Hammouda, I., Mikkonen, T., & Aaltonen, T. (2012). From proprietary to open source—Growing an open source ecosystem. *Journal of Systems and Software*, 85(7), 1467-1478.
- Kittlaus, H. B., & Clough, P. N. (2009). *Software product management and pricing: Key success factors for software organizations*, Berlin Heidelberg: Springer.
- Robertson, D., & Ulrich, K. (1998). Planning for product platforms. *Sloan Management Review*, 39(4), 19-31.
- Schuur, H. W. v. d., Jansen, R. L., & Brinkkemper, S. (2011). The power of propagation: On the role of software operation knowledge within software ecosystems, *Proceedings of the International Conference on Management of Emergent Digital EcoSystems*, San Francisco, CA, USA, 76-84.
- Takeda, H., Veerkamp, P., Tomiyama, T., & Yoshikawa, H. (1990). Modeling design processes. *AI Magazine*, 11(4), 37-48.
- van de Weerd, I., Brinkkemper, S., Nieuwenhuis, R., Versendaal, J., & Bijlsma, L. (2006). Towards a reference framework for software product management. *Proceedings of the 14th IEEE International Requirements Engineering Conference*, Minneapolis/St. Paul, MA, USA, 319-322.