Bridges and Barriers to Hardware-Centric Software Ecosystem Participation – a Case Study

Krzysztof Wnuk[a,*], Per Runeson[a], Matilda Lantz[a], Oskar Weijden[a]

[a] *Lund University, Dept. Computer Science, Box 118, SE-221 00 Lund, Sweden*

Abstract

Background. Software ecosystems emerged as means for several actors to jointly provide more value to the market than any of them can do on its own.
Objectives. This work aims at studying barriers and bridges to participation and governance activities in an ecosystem with substantial hardware dependencies.
Method. We conducted an interview-based case study of an ecosystem around Axis' network video surveillance systems, interviewing 10 internal experts and 8 external representatives of 6 companies, complemented by document studies at Axis.
Results. Major bridges to the ecosystem include end customer demands, open and transparent communication and relationship as well as internal and external standardizations. Barriers include the two-tier business model, entry barriers and execution performance issues.
Conclusion. Our results suggest that ecosystem leaders should share their sales channels with the ecosystem participants and focus on good communication and relationships as the dominant factors for the ecosystem participation. Moreover, we report that internal and external standardization can play a dual role, not only ease the development but also enable additional sales channels and new opportunities for the ecosystem participants. Ecosystem leaders should also focus on the performance and promptly reply to new functionality requests from the ecosystem participants.

1. Introduction

The software business landscape has recently been impacted by several fundamental changes. The independent effort of software vendors on creating software products was transformed into collaborative interplay of several actors, moving the critical mass responsible for the success of a product into the management of relationships rather than the quality and quantity of development effort [1, 2]. The traditional roles in software development are being redefined and, according to Manikas and Hansen [3], software ecosystems are taking software development outside the borders of a single software company. Additional value is created and new business models are emerging when companies are starting to compete on new levels [3] and rely on components and infrastructure from third party vendors or suppliers [1]. Relationships between these

*Corresponding author

actors shape the product software landscape into a software ecosystem where they collaboratively create competitive value [4].

The area of Software Ecosystems (SECO) is a relatively new field of research and a collective theoretical foundation is starting to emerge [1, 4, 3]. SECO was identified as an emerging topic within the software engineering research [2]. In a SECO, the surrounding platform, reasons for participation, governance activities and decisions of the platform leader influence everyone involved [5]. By understanding the factors behind participation of their software ecosystem, platform leaders could be better equipped to facilitate a successful software ecosystem [5].

Several authors studied business ecosystems based on or significantly dependent on software [6, 7, 8, 9, 10] but not many authors focused on incentives and reasons to join business ecosystems [11]. Furthermore, bridges and barriers to join and remain with a software ecosystem were studied by several authors [12, 13, 14, 13, 15, 16, 17], who do not explicitly focus on a hardware-centric software ecosystem. Moreover, only five out of 90 publications about software ecosystems identified by Manikas and Hansen [18] were considered hardware-centric. Finally, incentives and hinders were broadly studied in software engineering, [19, 20, 16, 17, 21, 22], yet not fully explored for hardware-intensive software ecosystems.

In this paper, we report an empirical exploratory case study that investigates the ACAP (Axis Camera Application Platform) ecosystem of Axis Communications AB (Axis). Axis is a company producing video surveillance cameras, i.e. focusing on hardware and embedded software, which is now showing an increasing interest in their software ecosystems. Moreover, Axis does not sell its products directly to end customers but via partners, which integrate the cameras into complete surveillance systems by configuring and extending Axis' software.

This study focuses on exploring the reasons and benefits for participation and hesitations and drawbacks of not joining the ACAP ecosystem. The ACAP ecosystem can be characterized as hardware-centric software platform ecosystem (please refer to Section 3 for a detailed classification) because the main source of revenue for Axis is camera sales, while software is gaining importance and could in the future be a large part of the revenue stream. The purpose is reached by answering the following research questions:

RQ1 Why do third party developers join and participate in the ACAP ecosystem?

RQ2 What makes third party developers hesitant towards joining the ACAP ecosystem?

RQ3 What are the causes and effects of the main reasons and benefits?

The paper is structured as follows: Section 2 presents related work while Section 3 presents the case company. We outline the methodology used in this study in Section 4 including validity discussion. The study results are outlined and disucssed in Section 5. We conclude the paper in Section 6.

2. Related Work

Software ecosystems is related to business ecosystems, an economic and social landscape supported by a foundation of interacting organizations where competitors often start collaborating and thus create and support "mutualism" [23]. In this network of interacting organizations, companies governing their business ecosystem need to properly identify and focus on the "keystone species" as they ensure the health of their business ecosystems and help to achieve their strategic goals and benefits [24]. Moreover, open negotiation environment can empower the potential of business ecosystems by negotiating alliances of loose networks of suppliers, distributors and outsources makers of products and providers of services [25].

Software ecosystems could also be created by expanding software product lines beyond their platforms and their organizational boundaries [19], thus creating benefits from the synergy of mass customization and accelerated open innovation. The existence of a large set of customers with a real reason to extend the platform, good development environments and stable interfaces are among the key ingredients that allow this transition [19].

Several authors studied business ecosystems based on or significantly dependent on software. Mäkinen and Dedehayir [6] listed many examples of business ecosystems, among others an mobile phone ecosystem [7], the Internet ecosystem [8], Cisco's business ecosystem [9] and Deutsche telekom's open innovation ecosystem [10]. Their review [6] focuses on members of business ecosystems and their roles, the evolution of business ecosystems, the dynamics of ecosystem changes and firm strategies in evolving ecosystems. However, only one of the reviewed by Mäkinen and Dedehayir articles focuses on incentives and reasons to join business ecosystems. Cusumano and Gawer [11] advocates that preserving the compatibility of the platform and maintaining the platform leaderships provide incentives for complementors to remain a part of the ecosystem.

In their review of software ecosystems, Jansen and Cusumano classified only two (Android and iOS) out of 19 software ecosystems as hardware-centric [1]. Moreover, out of 43 ecosystems identified by Manikas and Hansen from 90 publications [18], only five could be considered as hardware-centric: Linux Kernel [26] (a study of software evolvability), App Store [27] (a study of consequences of architectural decisions on openness), Nokia Siemens Networks [28] (a study of hybrid revenue models), Symbian [29] (a study of propagations of API changes among clients) and US DoD (a study of transfers between entities [14]). To the best of our knowledge, this paper presents the first study focusing on the bridges and barriers to a hardware-centric software ecosystem participation.

Several authors reported empirical studies in software ecosystems. Among these studies are: clusters of ecosystems based on the Ruby programming language [30], the Hadoop ecosystem [31], App Store, Marketplace, Play [32], SAP, Open Design Alliance, Eclipse Foundation [33], GX [13] just to name a few [18]. However, most studies focused on small or medium-sized ecosystems and to our best knowledge, no study investigated hardware-centric ecosystems. Moreover, Santos et al. discussed the emerging role of SECOs as a research topic in software engineering [2] highlighting that the SECO research is currently concentrated around open source software, ecosystem modeling and business issues and not about bridges and barriers. Case studies about software ecosystems are conducted with varying levels of rigor [2]. To summarize, there is lack of empirical studies focusing on large hardware-centric ecosystems [3] conducted with high methodological rigor.

Bridges and barriers to join and remain with a software ecosystem were studied by several authors. The intellectual property management process can be an important reason to join an ecosystem [12]. At the same time, Jansen and Cusumano stressed that without the "plankton" (a potential market of sufficient size) it is risky for companies to join ecosystems [4]. The number of new players joining an ecosystem is highly correlated with its productivity [13] and robustness of software ecosystems [14]. However, the presence of a clear dominator in an ecosystem could be a discouraging factor for newcomers [13]. Jensen *et al.* concluded that fast response to newbie posts is correlated with their future participation in an open source project [15]. Unstable or unclear architecture was mentioned as one of the barriers to join an open source project [16]. Peng *et al.* stressed the role of competition and reduced downstream competition in the formation of enterprise software ecosystems [17].

Incentives and hinders were broadly studied in software engineering, from subjects in experiments [20] to open source projects [16]. Bosch identified four success factors to make a platform an interesting choice for third party developers [19]. The first one is the existence of a large set of customers with a real reason to extend the platform functionality with third party applications [17, 19, 21] which ensures the compatibility [22] and is a prerequisite to access this aforementioned customer base. Third party developers need to know that the platform leaders' customers actually have a need for the extended functionality of their applications [19], since the customers purchase not necessarily was made with third party applications in mind. The second success factor is simplified development [19] which enables third party developers to maximize their profit [21], by reducing the costs to attain the pool of customers. Good development environments and stable interfaces are also important factors attracting developers [19]. The third factor is seamless customer experience [19]. Ceccagnoli *et al.* highlighted the value of software interoperability and seamless integration in the eyes of the customers for example through the same user experience framework [22]. The fourth factor is a viable market channel which allows potential customers to be exposed to innovation fostered from new participants [22].

3. Case description

Axis was founded in 1984 as a company delivering print servers, but has over the years moved to become the market leader within network video and network video surveillance cameras [34]. Today Axis profits are mainly related to sales of camera units. The company provides network video solutions for professional installations featuring products and solutions that are based on innovative and open technology platforms. As the amount of software in the video surveillance cameras continues to increase and gains more importance, Axis sees potential in exploring and developing their hardware-centric software ecosystem. The company is based in Lund but has offices in 41 countries, partners in more than 179 countries and has 1400 employees. In 2012 Axis annual turnover was approximately 4 billion SEK [35].

Axis was founded upon a distinct two-tier business model with indirect sales. Identification and maintenance of partnerships is considered as a scalable solution for sales, but requires several different actors such as distributors, system integrators and technology vendors to provide complete solutions to end customers. Axis has four partner programs – Channel Partner (CP), Architecture & Engineering (A&E) Partner, Technology Partner (PTP) and Application Development Partner (ADP), initiated in 2000 – which allow access to the Axis Camera Application Platform (ACAP) ecosystem.

There are currently three tiers in the Axis' program: (1) member of application development service (ADS), (2) application development partner (ADP) and (3) gold application partner (Gold ADP). Requirements for joining the program are low, but to advance on to higher levels actively engaged with Axis, companies have to prove that their solutions generate a certain amount of camera sales. These companies have to successfully integrate their commercial applications with a significant portion of the Axis product range. To reach the Gold ADP level Axis need to be selected as a preferred network video hardware vendor by a company. On this level, volume channel license requirements are fulfilled, roadmap information is shared, development resources are dedicated for integrating new Axis products and features, the company is regularly engaged with Axis business development managers [34].

The ACAP ecosystem is based on an open application platform that enables development of third party applications. These applications can be downloaded and installed on Axis' cameras and video encoders. The platform was launched in September 2009. The main reason was to extend the functionality of the camera to meet evolving end user needs [36]. In order to enable and facilitate development towards the platform Axis also provides: an API, a Software Development Kit, a compatibility tool and copy protection tool. As the main source of revenue for Axis remains camera sales, we consider this ecosystem as hardware-centric.

We classify the ACAP ecosystem based on the classifications proposed by Jansen and Cusumano [4] and Bosch [19]. The base technology aspect the ACAP ecosystem can be categorized as a *software platform* since ACAP applications are installed on physical cameras. The ACAP is an *application-centric* ecosystem based on the software platform that have already achieved success in the marketplace without an ecosystem, i.e. the platform offers customer value without third party applications [19]. The ACAP applications are domain specific and extend the functionality offered by the platform. Related research made the same assessment regarding an ecosystem of embedded software in the car industry [37] which supports our classification of ACAP as an application-centric ecosystem. Regarding accessibility, in order to get access to the ACAP ecosystem a company has to be a member of Axis Application Development partner (ADP) program, free of charge with low membership requirements [38] but with sales requirements to advance to higher levels. Therefore, the ACAP ecosystem can be considered as screened but free.

Regarding the extension market, the ACAP ecosystem has a list of extensions available on Axis' website and it is entirely controlled by Axis [36]. Axis is not handling any sales or transactions and does not offer any joint way of purchasing third party applications, which forces third party developers to sell their software in other ways. Customers of the ACAP applications are redirected to websites of companies developing ACAP applications in order to download or purchase them. This flow of sales is included in red in Figure 1. Optionally, Axis can offer a licensing system which could also be seen as a part of the ex-

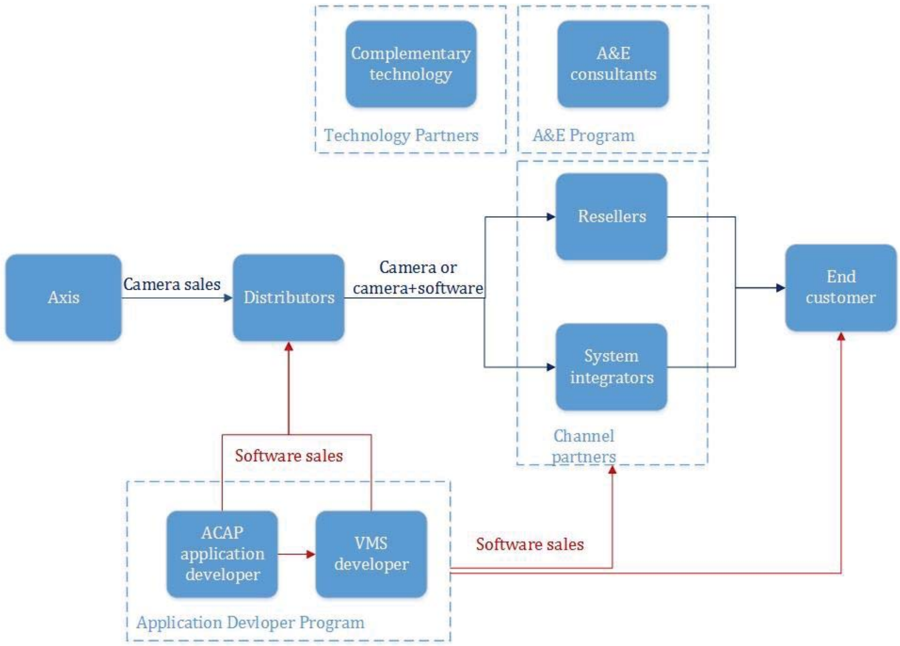

Figure 1: The software ecosystem surrounding the ACAP.

tension market. Finally, no direct network effects are observed in the ACAP ecosystem. Currently, the ACAP developers do not generally benefit from sales made by other third party developers. A purchase of an application by an end customer does not raise incentives of buying another application. The technical capabilities of the cameras generally allow one application to be run at once. This blocks the co-creation of value through utilizing each other's applications and data.

The ACAP ecosystem involves several actors, see Figure 1. Axis is the platform leader which has the biggest influence on the decision about the ecosystem. The main group of external actors constitute Video Management System (VMS) developers. They develop external products, running on servers or similar, and most of them receive image output or control cameras. Some of these video management systems utilize functionality of ACAP applications through incorporating them into their own programs. System integrators and resellers (both small local and large global) are also among the actors and they are involved in the channel partners program. This group is classified as vendors because they are making money on selling products produced by the software ecosystem. End customers are the largest group of ecosystem participants (and potential buyers of the ACAP applications) who indirectly influence the evolution of the ecosystem via their requirements and needs. However, they are not directly involved in the growth or strategic decisions in the ACAP ecosystem. Finally, among other participants are distributors, although most of them incorporate software into cameras before selling them and therefore they cannot be classified as vendors [18].

4. Case Study Design

This study was initiated by a need from Axis to explore their software ecosystem from the perspective of software developers surrounding the platform. Due to limited related work that explicitly focuses on the studied phenomena, see Section 2, we decided to conduct and exploratory case study [39]. The business aspect was excluded at the early stage. A competitive blend of researchers and practitioners was formed to match the requirements of the studied case and fully utilize the experiences of both. As the case company is relatively new in software ecosystems, an exploratory case study method was selected [39] as suitable for exploring the studied phenomena. The survey strategy [40] was excluded due to the low respondents rate risk and the multiple case study strategy (in several ecosystems) was not possible due to lack of access to more hardware-centric ecosystems.

4.1. Research Process

This study followed the case study process proposed by Runeson *et al.* [39]. The *flexible* nature of the case study strategy [39] allowed for iterations between the research process steps. The work process and steps are visualized in Figure 2 and are outlined below:

1. Pre-study, see I in Figure 2
2. Case study design, see II in Figure 2
3. Preparation for and data collection, see III in Figure 2
4. Analysis of the collected data, see IV in Figure 2
5. Reporting, see V in Figure 2

We conducted a pre-study (see I in Figure 2), in order to gain initial information about the research area of software ecosystems. In the pre-study (step I in Figure 2), company specific literature and related work were studied. To explore the practical context and relate it with identified publications, ten exploratory interviews were conducted among practitioners knowledgeable in the ACAP ecosystem, see Table 1. This enabled knowledge transfer and better understanding of both software ecosystem theory and the situation at Axis through continuous iteration between the empirical findings and published knowledge [41].

In the following case study design phase (step II in Figure 2), we set the objectives of the study and formed the research questions. The scope of the case study was defined as Axis together with the group of companies currently developing towards the ACAP, broadened by the companies that decided not to develop toward the ACAP (companies C and E in Table 2).[1]

In the preparation and data collection phase (step III in Figure 2), we collected data through eight semi-structured interviews at six companies. Interviews are classified as a direct data collection technique [42] where the researcher is in direct contact with the subjects. Hence the researcher can, to a large extent, control what data is collected, how it is collected and in what context [39].

[1] The companies listed in Table 2 are sample companies and thus do not constitute the full list of the ACAP ecosystem participants.

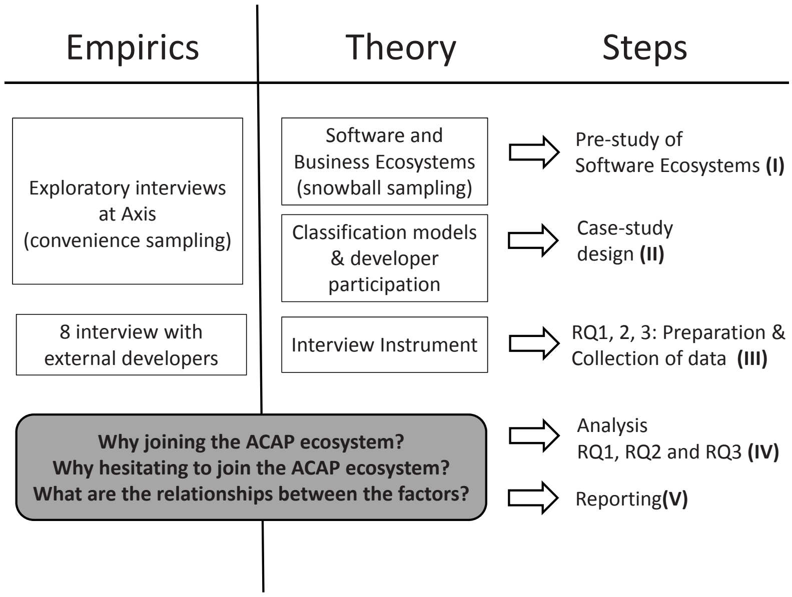

Figure 2: Visualization of the iterative work process where empirical and theoretical studies are performed in parallel. The connection to Runeson *et al.*'s [39] proposed steps are presented to the right in the figure.

The interview results were analyzed (step IV in Figure 2) using coding and tabulation, see Section 4.4 and using document studies supported by exploration of related work.

4.2. Case and subject selection

Axis was selected as a case company due to the following reasons: (1) it is a large company that operates globally, (2) it develops embedded systems and provides a case of a hardware-centric software ecosystem, (3) it does not have any direct sales of the products to end customers, and (4) the end customers do not get directly involved in the development or strategic decisions about the ecosystem.

The pre-study interviews gave supporting knowledge and empirical evidence for selection of participants in the study. The interviewees were identified by convenience sampling through practitioner recommendations and through snowball sampling by recommendations from the initial interviewees [39].

The empirical data were collected through eight semi-structured interviews in six different companies, presented in Table 2. Participants were selected based on: if they had an ACAP application available, if they were involved in the decision process to develop towards the ACAP, and if the have both technical and business expertise in the area. Several interviewees had both business and technical expertise as they worked for relatively new and thus small companies. Following Axis advice, two companies that shown an interest towards the ACAP ecosystem, but were not developing ACAP applications, were also included in the study (companies C and E in Table 2).

Table 1: Internal interviewees at Axis

Work title
Global Partner Manager
Product Manager Solution & Integration
Programs & Partner marketing
Global ADP engineer, Partner Management
Director, System & Services
Senior Engineer, Video Hosting System
Business Development Manager, Business Development, Northern Europe
Product Manager API & Components
Global Partner Manager, Partner Management
ADP program manager

4.3. Data collection procedures

The interview questions (available in Appendix A) were based on information obtained during the pre-study presented in Section 4.1. Semi-structured interviews were seen as the most appropriate due to the exploratory goal of the study. Semi-structured interviews can help to ensure that common information between interviewees is collected and allow the interviewer to dig deeper when required [43]. All planned interview questions were included in an interview instrument (presented in Appendix A) which was used as a template during the interviews. The interview instrument was reviewed prior to conducting the study. Two researchers were present during all interview sessions, conducted in English. This allowed one interviewer to focus on asking questions while the other one could take notes, reflect and ask follow-up questions. All interviews were recorded and transcribed.

4.4. Analysis procedures

The recorded and transcribed interviews were analyzed and answers related to RQ1, RQ2 or RQ3 were tagged by unique ID:s. Since a structured approach is important when conducting a qualitative analysis, the authors used tabulation where the coded data was arranged into tables (later merged into Table 3), a method suggested by Runeson *et al.* [39].

According to Eisenhardt [44] it is important to first get familiar with the case in order to let the patterns emerge before pushing to generalized patterns. Therefore the analysis first focus on factors connected to the companies included in the case before trying to lift it to more general conclusions. An analysis of the context and underlying factors of the companies, interviewees and their answers was conducted in order to identify patterns or connections, a technique that is called explanation building [39]. Finally a cause and effect analysis of the identified reasons (RQ3) was performed in order to reveal underlying factors affecting the findings.

4.5. Validity analysis

Runeson *et al.* [39] distinguish between four types of validity, construct validity, internal validity, external validity, and reliability, which we analyze below.

Construct validity refers to what extent the studied operational measures represent what the researchers planned to investigate. A threat towards construct validity is if, for example, the interview questions are not interpreted in the same way by the researchers and the interviewees [39]. In order to minimize this risk, we piloted the interview questions on three employees at Axis and two independent researchers in two iterations before presenting them to the interviewees. To avoid quotation becoming out of context, two researchers performed transcription and coding where differences were discussed and conflicts were resolved. Finally, the results of the study were presented and discussed with the participants at a workshop.

Internal validity is related to investigations of causal relations. When examining if a factor affects a factor of investigation there is a risk that this factor of investigation is also affected by a third factor [39]. Members of a software ecosystem are often described as closely affecting each other in complex networks [19]. We reduced this threat by: (1) providing extensive classification in order to map possible factors and how they affect each other and (2) collecting data from several sources, internal at Axis and from external companies. The resulting triangulation allowed to reduce internal validity threats [39].

External validity is concerned with to what extent it is possible to generalize the findings, and to what extent the findings are of interest to other people outside the investigated case [39]. There are researchers that argue that case studies cannot be generalized [45] while others argue that a generalization can be made as long as an extensive characterization is performed along with analysis [39]. Runeson *et al.* [39] suggest that the intention of case studies is to enable generalization where the results extend to cases which have common characteristics and hence for which the findings are relevant. We provide classifications of the case company in order to facilitate generalization and comparison with other cases.

We believe that the identified bridges and barriers can be applied to a broader selection of ecosystems. However, the selection of the ecosystem under study may impose a threat to the generalization of the results. As mentioned in Section 2, hardware-centric ecosystems are rarely reported in the literature as we found only seven related examples. Moreover, the ACAP ecosystem shares more characteristics of an *application-centric* software ecosystem based the software platform (e.g. App Store [27]) and therefore may not be a good representative of the *operating system-centric* ecosystem, such as examples reported in the literature (Symbian [29], Android and iOS [1], Linux Kernel [26]). At the same time, ACAP constitutes an interesting example of a hardware-intensive ecosystem that is *application-centric* rather than *operating system-centric*. To what extend our findings could be generalized to larger ecosystems, e.g. the US DoD [14] or Nokia Siemens Networks [28] ecosystems remains to be investigated in future work.

Reliability refers to the aspects of to what extent the data and the analysis are dependent on the researchers [39]. In order to increase reliability, an interview instrument was created to provide the interviewers with guidance and making sure that all relevant aspects were covered in all interviews. Two authors were present at all interviews in order to reduce bias. The study procedure was created before execution, as reported in Section 4.1.

5. Results

We identified 14 *bridges* and 10 *barriers* of software ecosystem participation. Bridges were divided into *reasons* and *benefits* while barriers were divided into *hesitations* and *drawbacks*. We consider a reason more high level than a benefit since a reason could embed a benefit but not vice-versa. Moreover, a reason is usually of higher importance than a single benefit. Therefore, we chose to separate them to make comparisons easier.

The main reason to distinguish between hesitations and drawbacks was to be able to investigate the process of joining an ecosystem when most of our respondents were already involved in the ecosystem. We consider drawbacks as barriers visible also after joining the ecosystem that our respondents could have found solutions to some of them. For example, lack of documentation might currently not be a drawback, but could have been an hesitation when a company was evaluating whether to develop towards the ACAP platform or not. A hesitation could also capture smaller issues that the respondents might not consider as drawbacks after joining the ecosystem. The results are presented below and summarized in Table 3.

5.1. Reasons and Benefits for software ecosystem participation (RQ1)

The two main reasons for participating in the software ecosystem are *end customer demand (Re1)* and the *communication and relationship (Re2)* with Axis as the platform vendor. Three out of four companies (A, B, D) that currently develop towards the ACAP expressed *end customer demand* as a reason to participate in the ecosystem, see Table 3. Company A received a clear demand from a customer that their video analytics solution should be integrated with Axis' cameras. Company B initially approached Axis with another video analytics application which Axis was not interested in. Instead Axis suggested company B to develop another application which Axis had identified a need for. Finally, according to respondent D1, they received demand from end customer via their sales department. However, respondent F1 did not mention end customer demand as a reason for joining the ACAP ecosystem. One reason for this could be that the company develops video management systems, not video analytics.

The *communication and relationship* with Axis was mentioned as a reason for developing towards the ACAP in companies A, B and F. All companies in this study that have an ACAP-product available did have a good relationship with Axis prior to starting developing towards the ACAP. Three out of four (A, B, D) companies emphasized transparency and openness in communication. Company E considered having opaque relationship with Axis, findings it difficult to talk to the right people and to get clear answers about technical specifications, future development and end customers. As both companies not developing towards the ACAP (C and E) reported challenges in communication and relationship, this could indicate that good communication facilitates adopting the Axis' business model (see Section 5.2).

Respondent A1 mentioned creating an *open environment* as one of the reasons to join the ACAP software ecosystem. Six years ago company A discovered that Axis is providing an open SDK and decided to start developing applications based on this SDK. Company B considered *geography* as the reason to join ACAP as they also are located in Lund and thus company B reasoned that the communication and collaboration would be much more easier due to this proximity. Company D mentioned *future possibilities* (Re5) among the reasons to join the ACAP ecosystem as they project that their applications are going to be used in other sectors, for instance the retail sector and for that they need a stable platform (ACAP). The same company mentioned *marketing* and in particular their presence as an ACAP partner on the Axis' website as an opportunity to create more relationships with potential partners and customers. Finally, Company F mentioned that joining the ACAP ecosystem is a *low risk* endeavor (Re7) since the ACAP platform is running on Linux and Axis had the largest market penetration in North America so the potential reward to company F was considered as very high.

Internal (Be1) and *external standardization* (Be2) were found to be the two main potential benefits of participation in the ACAP ecosystem. Companies A, B and D mentioned internal standardization (enabling the same code to be used in the majority of Axis' cameras) and companies D and E mentioned external standardization (allowing communication between the camera and external software) as potential benefits. Both companies A and B experienced a decrease in efforts put into modifying their code when developing their applications towards the ACAP. This result supports previous research on the importance of simplified development and stable interfaces for third party developers [19]. These two benefits are not experienced by companies C and E mainly because they have not gone through the process of developing the ACAP applications.

External standardization would allow company D to sell their analytics solution separate from their video management system. Moreover, through a standard platform, companies compatible with Axis' cameras would also be compatible with their video analytics products. This would create a wider network of potential end customers and help with the *marketing* activities, mentioned as a benefit earlier. Participation in the ACAP ecosystem could also reduce the need of local support due to easier installation and increase compatibility.

External standard is also seen as a benefit that helps company E to focus their work on camera and save implementation resources for each VMS. The benefits for company A from external standardization are only partial because their application runs partly on the ACAP and partly on a server with the latter facing other systems. Finally, solutions developed by companies B (business intelligence solutions) and F (VMS) do not communicate with external software and therefore they do not benefit from a standard for communicating with external software.

When company A1 became interested in joining the ACAP ecosystem, they were a quite small company and for the an option to team-up with a large player like Axis and *piggyback* on Axis (Be3) was a clear benefit. Another mentioned benefit was *easier installation* (Be4) of the ACAP-based applications with the cameras than the PC based solution developed earlier by company B. Company D mentioned that *easier installation* enabled more system integrators to work with them and increased the speed of installations. Related to the easier installation is the benefit of *less infrastructure* (Be5) that allows company D not to worry about the service of the infrastructure (cameras and the associated IT infrastructure) since this part is provided by Axis retailers and integrators.

Scalability (Be6) of the video cameras infrastructure was mentioned as one of the benefits by company F. If their customers wants to add additional cameras, company F does not need to be responsible for the central server and network upgrades as this is the part Axis takes care of. Moreover, each camera adds enough resources to the network system as a whole and has the capability of completely be VMS independent. This in turn decreases the need for additional server infrastructure and *resources* (Be7) and lowers the total power consumption of the provided solution making it more environmentally friendly.

5.2. Barriers to software ecosystem participation (RQ2)

The two main barriers against participation in the ACAP ecosystem are *Axis' business model* (see hesitation *He1* in Table 3) and *execution performance* (see drawback Dr1 in Table 3).

Both companies that currently do not develop towards ACAP mentioned Axis' business model as one of the main hesitations. For company E, acquiring information about end customers from Axis in the search for new potential customers is a significant issue. They prefer to work directly with system integrators or VMS companies through an OEM (Original Equipment Manufacturer) business model while Axis' two-tier business model is focused on large distributors. Company C also prefers the OEM business model as they do not perceive that the security market is interested in purchasing software after a camera is bought, but rather at the same time, making bundling of software and camera into one product more effective.

Since companies A, B, D and F had an existing product within the industry prior to joining the ACAP ecosystem, they had established sales challenge within the industry that could potentially be reused for marketing the newly developed ACAP application. We believe this was the main reason why these companies did not mention Axis' business model as one of the a barriers.

Technical features were also mentioned as a hesitation towards developing ACAP applications, see *He2* in Table 3. Our respondents mentioned several technical limitations, e.g. that the current platform allows only one application to be run at a time, limiting the technical capabilities of potential ACAP applications and their multitasking or that the audio streaming needs to be done via HTTP not directly from the cameras. On the other hand, company C mentioned *unclear roles* of cooperation with Axis as one of the hesitations (He3) to join the ACAP ecosystem. In particular, the roles in the sales chain, in contributions to the ACAP platform, in forming strategic alliances and benefits from the partnership program were stressed as important to be clarified. Company E listed lack of information about large customer installations shared by Axis (see *He4* in Table 3) as one of the hesitations to join the ACAP ecosystem. If this information is shared by Axis, company E could perform market and opportunity analysis and plan to develop or upgrade their applications to target these large installations and minimize the marketing efforts.

The *strict rules of Application Development Program (ADP)* were considered by Company E as a hesitation (He5) mainly due to high requirements to advance to higher levels in the program. Axis requires companies to provide a prove that their solutions developed by them generated a certain amount of sales for Axis and that their engagement with Axis remains active after that. Moreover, only companies can be members of the ADP program, limiting the participation of individual developers and therefore possibilities to increase the quality of the platform by frequent by small bug fixes and improvements suggestions. Finally, company F mentioned that searching for or retrieving a video produced by an Axis camera with the help of the current software platform is not straightforward and therefore creates an additional *maintenance* effort for the developed ACAP applications, see He6 in Table 3. Interestingly, installation does not seem to be a hesitation since easier installation was earlier mentioned as a benefit by two companies.

No barriers when adapting the application to the ACAP ecosystem were identified among companies A and B. The main reasons for that could be that both companies were involved in development of embedded solutions in Axis' cameras prior joining the ACAP ecosystem. Furthermore, Companies A and B had already developed embedded applications to Axis cameras before joining the ACAP which means that they most probably already had aligned their business model with Axis'.

All companies developing ACAP applications and company C mentioned *performance* as the main drawback (Dr1) of developing applications towards the ACAP ecosystem. Lack of dedicated processor power to third party applications (mentioned by companies A and B) low efficiency (pointed out by company F) and low capacity and performance variability (according to respondent D1) were listed as the main performance drawbacks. These performance drawbacks are associated with the *technical features* hesitation He2 mentioned earlier.

Next, *lacking debugging capabilities* (mentioned by respondent D1) was listed as a drawback of ACAP participation that negatively impact the efficiency of ACAP application developers. Company D reported dedicating extensive time to debug their ACAP application. This problem is also associated with *information-gap* drawback (Dr3) as Axis could have improved the debugging capabilities documentation. Moreover, the information-gap considers also seamless access to information about Axis' customers, better overview of roles and rules of ecosystem participation and benefits from advancing the ADP program (see also the results regarding hesitation He4). Finally, *lack of uniform camera compatibility* is a drawback (Dr4) that requires additional development effort when porting ACAP applications between various camera models.

5.3. Causes and effects of the main reasons and benefits (RQ3)

To further investigate the studied phenomena and explore their relation to the identified in the literature context factors [4, 18], we performed a cause and effect analysis of the main reasons and benefits, using fish-bone diagrams [46]. Six contextual factors and their subgroups [4, 18] were used as starting points to find potential causes. The collected empirical evidence was confronted with related work, in order to identify specific underlying causes that affected the identified reasons. We performed this analysis for the main reasons and main benefits of joining the ACAP ecosystem. To increase the quality of our analysis, we excluded reasons and benefits mentioned by single companies and focused on summarizing findings that could be triangulated with views from another companies. For single company reasons and benefits, we provided additional information, if relevant, together with their presentation and description in Sections 5.1 and 5.2. The following factors were used as a starting points to find potential causes:

- Actors
- Base technology

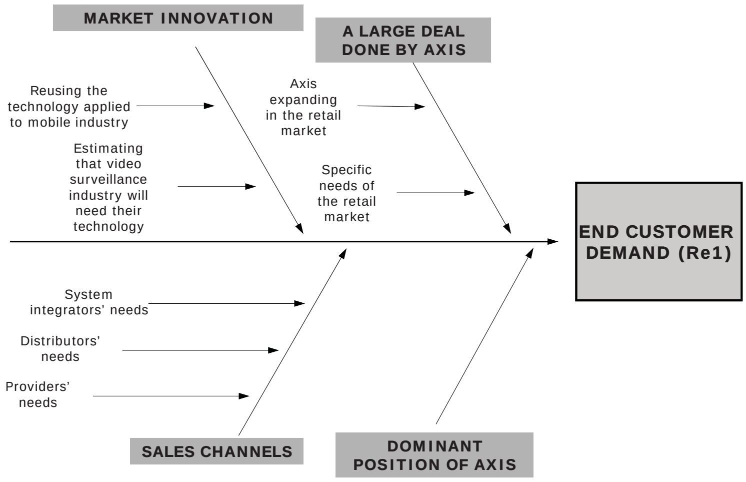

Figure 3: Causes analysis for end customer demand (Re1).

- Accessibility
- Extension market
- Network effects
- Competing platforms

Empirical data from both internal discussions and external interviews was combined with knowledge derived from related work in order to identify underlying causes that affected the identified reasons. The effect, i.e. the reason for development and benefits, were put on the right hand side of the fish-bone diagrams. The horizontal lines represent causes which in some cases are broken down into sub-causes (see Figures 4–6).

5.3.1. End customer demand (Re1)

For company A, the main cause of Re1 was actually a large deal done by Axis which sold cameras to one of the large retailers in the US. Company A was approached by this retailer with a specific request to develop video analytics solution based on hardware and firmware provided by Axis. The effects of this endeavor were not only increased sales but also creation of a product compatible with all cameras provided by Axis at that time that could be targeted towards other customers in this market segment and other customers of Axis. The main reason for company B was that they had a face detection solution (base technology) that could was first used with photos taken by mobile phones to make funny caricatures. Company B approached Axis and convinced them that

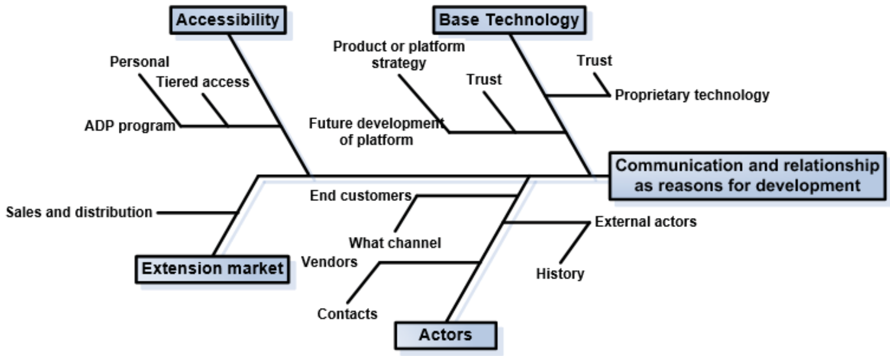

Figure 4: Causes analysis for communication and relationships as a reason for development (Re2).

their face detection solution should be put on video cameras. After some initial resistance, Axis found out that there are many potential customers for the face recognition component rather than the caricatures. The effect was that the customers received a face recognition solution that could be used for example to count the number of people entering a build and that was also compatible with the ACAP platform and could serve future customers. For company D, the ACAP helped to better use three out of four main sales channels (extension market) and get application needs from: system integrators, distributors, and providers. The sales department of company D saw increased demand for ACAP applications from the mentioned three sales channels. The observed effects of these demands where porting video analytics components developed by company D into the camera using the ACAP compiler. The result was the software analytics compatible with Axis cameras that did not have to be re-implemented and extensively re-tested .

5.3.2. Communication and Relationship (Re2)

The most dominant causes of communication and relationship turned out to be: (1) the need of trust in base technology, (2) accessibility to customers and information, (3) extending the market and (4) actors, see Figure 4.

Base technology: Developing towards the ACAP requires customizations towards the Axis proprietary technology and therefore requires trust in the base technology. Making investments in technology that cannot easily be redeployed is commonly called an asset specific investment [47]. Developers need to trust Axis (having the full control over the platform) in their future platform development plans and to accept them. The need for trust was identified as a possible cause for good communication and relationship which in turn could reduce risks in software ecosystems [48]. Developers without trust in Axis would probably be wary of Axis focusing on the basic camera features (since the ACAP is a small part of Axis' current business) rather than developing the platform, as indicated by Hagel *et al.* [49]. The perceived effects of this cause are an increased number of companies developing towards ACAP and a decreased number of dropout companies that discontinue developing towards ACAP.

Accessibility: Axis ADP program offers tiered access to business and technical information. However, the highest tier (Gold ACP) is not commonly reached by ACAP developers. Instead, personal relationships are utilized to receive information of the higher levels, thus causing communication and relationship to become a reason for access and development (Re2) and creating hesitations due to difficulties in accessing the information about the customers (He4). One visible effect is that the potential of the ACAP ecosystem in terms of participation is not fully explored, especially among universities and private contributors.

Extension market: We can confirm the related work viewpoint [19] that providing a viable market channel is a large influencing factor on developer's incentives to join an ecosystem. However, Axis does not provide a way of selling or distributing the ACAP applications. Therefore developers without an established customer base or a relationship with Axis would experience a more uncertain way to market, see also Section 5.3.4. The experienced effects is an extensive growth of small companies that succeed to use the market channels offered by ACAP, see also causes for end customer demand (Re1).

Actors: The developer's history with Axis (the platform leader) was found to be a cause to participate in the ecosystem. By aligning more products with Axis, the relationship is strengthened, providing third party developers with more information and support. The need of such support is enhanced by the fragmented customer base and the lack of a central marketplace. The interplay between Axis and the ACAP actors often creates a synergy effect, mentioned earlier when Axis was approached by company B offering a face recognition solution that could be made compatible with the ACAP platform and thus reached more Axis customers.

5.3.3. Internal Standardization (Be1)

Internal standardization was described as both benefit and reason for development towards the ACAP, see Table 3. The cause analysis for internal standardization is depicted in Figure 5. We identified the following causes: (1) competing platforms, (2) base technology, and (3) actors. The effects of internal standardization (reusing the same software on different types of cameras) are similar to code reuse in software product lines [37], making development of several configurations more efficient in terms of cost and risk [50].

Competing Platforms: For internal standardization, the fact that no competing platforms offer an internal standard for embedded software in their cameras was identified as an important cause for interest or participation in the ACAP.

Base technology: In response to different customer needs, Axis offers several camera product lines [51] with different hardware configurations and the ACAP is installed on these product lines. This is an important part of Axis' product strategy. An internal standard becomes important to third party developers as fewer configurations are needed to work with all type of cameras. This is an important change for Axis since previously, new cameras were not tested for compatibility with third party applications.

Actors: The customers (end customers or vendors) are fragmented by the type of camera they use or sell. This makes internal standardization important to reach a larger group of potential customers. Additionally the past relationship with Axis was deemed an important cause for participation, as developers who developed prior to joining the ACAP (without the internal standard), saw a greater benefit in joining the ecosystem.

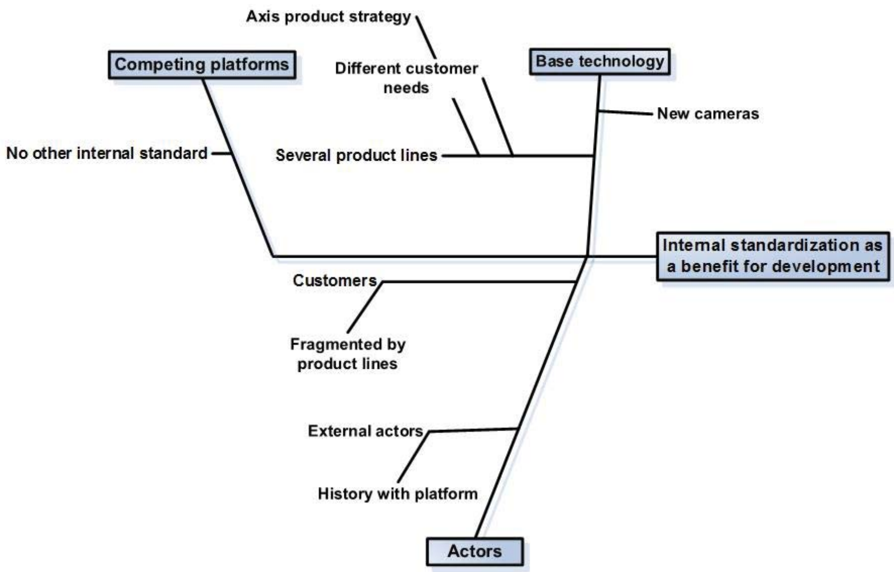

Figure 5: Causes for internal standardization as a reason for development.

5.3.4. External Standardization (Be2)

External standardization is the ability to more easily integrate ACAP applications with other software. In our case, external standardization was not fully implemented, but the surveyed developers saw a future potential of its implementation. We identified two causes and several factors affecting external standardization, see Figure 6.

Base technology: All Axis cameras require some type of software to manage the camera, such as a VMS. This software can either be embedded in the camera as an ACAP application or be provided externally. When ACAP applications provide more features than the standard camera does, the external software has to interpret the information from these applications in order to use it. This interpretation requires customization between the ACAP application and the external software. As a result, ACAP developers become dependent on external interpretation software. An external standard would reduce that dependence and the need of such customization.

Actors: The niched nature of the developed ACAP applications requires compatibility with other software components to provide a complete customer solution. Several of the identified causes of external standardization point at this compatibility. The amount of software versions (each having its own API), raises the entry barriers and increases the need for an external standard. The degree of fragmentation in the use of different VMS:s (or similar controlling software) depends on the industry. In order to reach customers who already have an installed ACAP solution, developers have to be compatible with that equipment. A higher compatibility with those systems may decrease the entry barriers and increase the potential customer base.

The ACAP developers are niched video analytics companies and are often smaller than the VMS developing companies. Therefore, their dependence to-

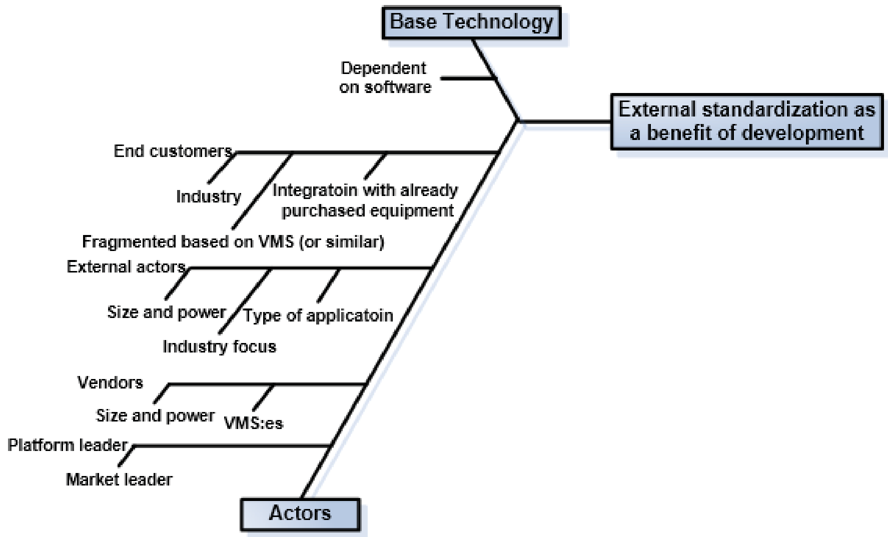

Figure 6: Causes for external standardization as a reason for development.

wards others might increase. Hence, the ACAP companies have to customize their solutions to fit every VMS whose customers they want to reach. However, some companies developing ACAP applications do not need external standardization. They provide their own VMS(s) or integrate their applications through other means. Finally, external standardization removes the dependence on the ACAP software and allows the companies to build their applications based on other platforms and solution that may provide better performance or serve specific customer needs or requirements.

6. Conclusions and future work

We have conducted a case study of the software ecosystem around the Axis surveillance camera application platform (ACAP). We reviewed Axis internal documentation, interviewed Axis employees and employees in six other companies participating in or relating to this ecosystem. The main goal for the study was to empirically explore the reasons and benefits for participation in the ACAP ecosystem and derive the causes and effects of them.

One of the more significant findings to emerge from this study is that end customer demands as well as good communication and relationships with the ecosystem leader are the main reasons for joining it (research question **RQ1**). The evidence from this study suggests that ecosystem leaders should stay aware that some customer demands or needs may emerge as a by-product of their activities or be delivered as solutions to uncovered demands by the ecosystem participants. Therefore, we suggest ecosystem leaders to stay open and share their sales channels with the ecosystem participants for potentially rewarding benefits. Our findings enhance our understanding of good communication and relationships as the dominant factors for the ecosystem participation [25] and for creation of an open environment for simplified development [19].

Taken together, the results regarding the benefits of the ecosystem participation suggest that internal and external standards are the most valuable benefits. Through an internal standard developers can reach a larger customer base at a lower cost. An external standard would decrease the need to individually collaborate with multiple partners to ensure compatibility and also allow the developers to reach more customers with less resources. At the same time, the possibility to piggyback on the ecosystem's leader, scalability, easier installation and decreased infrastructure demand should not be forgotten as potential benefits of the ecosystem participation.

Axis' business model and execution performance were identified as the main hesitations to join the ACAP ecosystem (research question **RQ2**). An implication of this is that the business model utilized by the ecosystem leader plays a dominant role in the ecosystem participation and can greatly decrease the number of participants. At the same time, the performance of the platform was considered as a main drawback after joining the ACAP ecosystem and developing applications for it which resulted in several request for technical features that the ACAP ecosystem leader should seriously consider.

The following conclusions can be drawn from the causes and effects analysis (research question **RQ3**). End customer demands received by the ecosystem participants may be caused by the ecosystem leaders' sales activities. Moreover, ecosystem leaders should stay open to solutions that may look like not demanded as they may actually represent tacit demands. On the other hand, the need for trust and a good relationship record are identified as the main causes for good communication and relationships. Finally, standardization (both internal and external) is caused by the different customer needs that need to be efficiently developed with fewer configurations and ensured compatibility with other applications. External standardization is also caused by the presence of other software vendors who use other VMS:s or similar controlling software.

The results presented in this paper have the following implications on the software ecosystems research and practice. They highlight main bridges and barriers for ecosystem participation in a hardware-centric software ecosystem where the main source of revenue remains hardware (video cameras). Just as experiences by the Volvo Trucks company [2], software has a great potential to become the main value contributor in hardware-centric software ecosystems as well as the key enabler for enhanced flexibility, on-the-fly updates and adaptive real-time functionality changes based on changing context factors.

Being aware of the study limitations, we would like to remind that the companies interviewed in this study are not all companies that develop ACAP applications. Moreover, the selection of the ecosystem under study may impose a threat to the generalization of the results as the ACAP ecosystem shared more characteristics of an application-centric software ecosystem than an operating system-centric software ecosystem, see Section 3. Moreover, we do not analyze if hardware imposes certain barriers and bridges and which of them are purely caused by software. Since software is easier to replicate and compete against than hardware, future work should focus specifically on the role of hardware in the hardware-centric ecosystems and explicitly identify which bridges or barriers are imposed by hardware and have (almost) nothing to do with software.

[2] http://www.swedsoft.se/Swedsoft_SRA_2010.pdf

Due to its exploratory nature, the current study was unable to identify all reasons and benefits as well as all hesitations and drawbacks. However, we believe that we took precautions in our study design to provide a representative sample of companies developing towards ACAP and therefore came up with as complete results as possible. Still, our findings remain limited by the use of an exploratory case study design and possible factors remain to be uncovered in future work.

Future work include replicating our study at another hardware-centric ecosystem. Moreover, we would like to further explore the bridges and barriers of ecosystems participation to unveil more aspects and confirm which bridges or barriers are specifically imposed by hardware and have (almost) nothing to do with software. Moreover, we plan to further investigate if the findings reported in this paper could be confirmed or refused at other companies where software plays primary role in the revenue stream generation.

7. Acknowledgements

The authors would like to thank Axis and the other companies in the ecosystem for providing the information to this study. Thanks to David Callele for excellent reviews of this paper. This work is funded by the SYNERGIES ICT framework project, Swedish National Science Foundation, grant 621-2012-5354.

8. References

[1] M. Cusumano, The Business of Software: What Every Manager, Programmer, and Entrepreneur Must Know to Thrive and Survive in Good Times and Bad, Business / The Free Press, Free Press, ISBN 9780743215800, 2004.

[2] R. Santos, C. Werner, O. Barbosa, C. Alves, Software Ecosystems: Trends and Impacts on Software Engineering, in: Software Engineering (SBES), 2012 26th Brazilian Symposium on, 206–210, doi:10.1109/SBES.2012.24, 2012.

[3] K. Manikas, K. M. Hansen, Reviewing the Health of Software Ecosystems – A Conceptual Framework, in: 5th International Workshop on Software Ecosystems (IWSECO), 33–44, 2013.

[4] S. Jansen, M. Cusumano, Defining Software Ecosystems: A Survey of Software Platforms and Business Network Governance, in: The 4th International Workshop on Software Ecosystems, 2012.

[5] A. Gawer, R. M. Henderson, Platform Owner Entry and Innovation in Complementary Markets: Evidence from Intel, Journal of Economics & Management Strategy 16 (1) (2007) 1–34.

[6] S. Makinen, O. Dedehayir, Business ecosystem evolution and strategic considerations: A literature review, in: Engineering, Technology and Innovation (ICE), 2012 18th International ICE Conference on, 1–10, doi: 10.1109/ICE.2012.6297653, 2012.

[7] R. Basole, Visualization of Interfirm Relations in a Converging Mobile Ecosystem, in: Mobile Business, 2008. ICMB '08. 7th International Conference on, 65–74, doi:10.1109/ICMB.2008.32, 2008.

[8] A. L. Zacharakis, D. A. Shepherd, J. E. Coombs, The development of venture-capital-backed internet companies: An ecosystem perspective, Journal of Business Venturing 18 (2) (2003) 217 – 231, ISSN 0883-9026, doi:http://dx.doi.org/10.1016/S0883-9026(02)00084-8, URL `http://www.sciencedirect.com/science/article/pii/S0883902602000848`.

[9] Y.-R. Li, The technological roadmap of Cisco's business ecosystem, Technovation 29 (5) (2009) 379 – 386, ISSN 0166-4972, doi:http://dx.doi.org/10.1016/j.technovation.2009.01.007, URL `http://www.sciencedirect.com/science/article/pii/S0166497209000157`, ¡ce:title¿Technology Management in the Service Economy¡/ce:title¿.

[10] H. G. G. R. Rohrbeck, K. Holzle, Opening up for competitive advantage How Deutsche Telekom creates an open innovation ecosystem, R&D Management 39 (4) (2009) 420–430, ISSN 1467-9310, doi:10.1111/j.1467-9310.2009.00568.x, URL `http://dx.doi.org/10.1111/j.1467-9310.2009.00568.x`.

[11] A. G. Michael A. Cusumano, Elements of Platform Leadership, MIT Sloan Management Review (2002) 1 – 10SMR079-PDF-ENG.

[12] K. Mizushima, Y. Ikawa, A structure of co-creation in an open source software ecosystem: A case study of the eclipse community, in: Technology Management in the Energy Smart World (PICMET), 2011 Proceedings of PICMET '11:, 1–8, 2011.

[13] S. Jansen, S. Brinkkemper, A. Finkelstein, Business Network Management as a Survival Strategy: A Tale of Two Software Ecosystems, in: Proceedings of the First Workshop on Software Ecosystems 2009, IWSECO '09, 34–48, 2009.

[14] J. D. McGregor, A method for analyzing software product line ecosystems, in: Proceedings of the Fourth European Conference on Software Architecture: Companion Volume, ECSA '10, ACM, New York, NY, USA, ISBN 978-1-4503-0179-4, 73–80, doi:10.1145/1842752.1842773, 2010.

[15] C. Jensen, S. King, V. Kuechler, Joining Free/Open Source Software Communities: An Analysis of Newbies' First Interactions on Project Mailing Lists, in: System Sciences (HICSS), 2011 44th Hawaii International Conference on, ISSN 1530-1605, 1–10, doi:10.1109/HICSS.2011.264, 2011.

[16] G. Von Krogh, S. Spaeth, K. R. Lakhani, Community, joining, and specialization in open source software innovation: A case study, Research Policy 32 (7) (2003) 1217 – 1241, ISSN 00487333, open source software;.

[17] P. Huang, M. Ceccagnoli, C. Forman, D. Wu, When do ISVs join a platform ecosystem? Evidence from the enterprise software industry, in: ICIS 2009 Proceedings - Thirtieth International Conference on Information Systems, Phoenix, AZ, United states, 2009.

[18] K. Manikas, K. M. Hansen, Software ecosystems – A systematic literature review, Journal of Systems and Software 86 (5) (2013) 1294 – 1306, ISSN 0164-1212, doi:http://dx.doi.org/10.1016/j.jss.2012.12.026.

[19] J. Bosch, From software product lines to software ecosystems, in: Proceedings of the 13th International Software Product Line Conference, SPLC '09, Carnegie Mellon University, Pittsburgh, PA, USA, 111–119, 2009.

[20] M. Höst, C. Wohlin, T. Thelin, Experimental context classification: incentives and experience of subjects, in: Proceedings of the 27th international conference on Software engineering, ICSE '05, ACM, New York, NY, USA, ISBN 1-58113-963-2, 470–478, doi:10.1145/1062455.1062539, 2005.

[21] F. Zhu, M. Iansiti, Entry into platform-based markets, Strategic Management Journal 33 (1) (2012) 88–106, ISSN 1097-0266, doi:10.1002/smj.941.

[22] M. Ceccagnoli, C. Forman, P. Huang, D. J. Wu, Cocreation of value in a platform ecosystem: the case of enterprise software, MIS Q. 36 (1) (2012) 263–290, ISSN 0276-7783.

[23] J. Moore, The Death of Competition - Leadership & Strategy in the Age of Business Ecosystems, Wiley, ISBN 9780471968108, 1999.

[24] M. Iansiti, R. Levien, The Keystone Advantage: What the New Dynamics of Business Ecosystems Mean for Strategy, Innovation, and Sustainability, Harvard Business School Publishing India Pvt. Limited, ISBN 9781591393078, 2004.

[25] L. Telesca, Towards Open Decentralized Self-Learning Negotiation Environment for Business Ecosystems, in: Communication Systems Software and Middleware, 2007. COMSWARE 2007. 2nd International Conference on, 1–5, doi:10.1109/COMSWA.2007.382419, 2007.

[26] L. Yu, S. Ramaswamy, J. Bush, Symbiosis and Software Evolvability, IT Professional 10 (4) (2008) 56–62, ISSN 1520-9202, doi:10.1109/MITP.2008.94.

[27] M. Anvaari, S. Jansen, Evaluating architectural openness in mobile software platforms, in: Proceedings of the Fourth European Conference on Software Architecture: Companion Volume, ECSA '10, ACM, New York, NY, USA, ISBN 978-1-4503-0179-4, 85–92, doi:10.1145/1842752.1842775, 2010.

[28] K. M. Popp, Hybrid revenue models of software companies and their relationship to hybrid business models, in: Proceedings of the Workshop on Software Ecosystems 2011, IWSECO '11, ISBN 978-1-4503-0445-0, 77–88, 2011.

[29] R. Robbes, M. Lungu, A study of ripple effects in software ecosystems (NIER track), in: Proceedings of the 33rd International Conference on Software Engineering, ICSE '11, ACM, New York, NY, USA, ISBN 978-1-4503-0445-0, 904–907, doi:10.1145/1985793.1985940, 2011.

[30] S. Shaheen, S. Jansen, On Clusters in Open Source Ecosystems, in: 5th International Workshop on Software Ecosystems (IWSECO), 19–32, 2013.

[31] J. Y. Monteith, J. D. McGregor, J. E. Ingram, Hadoop and its evolving ecosystem, in: 5th International Workshop on Software Ecosystems (IWSECO), 57–68, 2013.

[32] S. Hyrynsalmi, T. Mkil, A. Jrvi, A. Suominen, M. Seppnen, T. Knuutila, App Store, Marketplace, Play! An Analysis of Multi-Homing in Mobile Software Ecosystems, in: S. Jansen, J. Bosch, C. Alves (Eds.), Proceedings of the Fourth International Workshops on Software Ecosystems, vol. 879 of *CEUR Workshop Proceedings*, CEUR-WS, 59ñ72, 2012.

[33] J. v. Angeren, J. Kabbedijk, S. Jansen, K. M. Popp, A Survey of Associate Models used within Large Software Ecosystems, in: 3rd International Workshop on Software Ecosystems, Brussels, 1–13, 2011.

[34] A. C. AB, About Axis Communications, http://www.axis.com/corporate/about/index.htm, 2013.

[35] A. C. AB, Annual report 2013, http://www.axis.com/files/annual_reports/2012annual_eng.pdf, 2013.

[36] A. C. AB, Participation in ACAP, http://www.axis.com/corporate/press/industry_news/article.
2013.

[37] U. Eklund, J. Bosch, Introducing Software Ecosystems for Mass-Produced Embedded Systems, in: M. Cusumano, B. Iyer, N. Venkatraman (Eds.), Software Business, vol. 114 of *Lecture Notes in Business Information Processing*, Springer Berlin Heidelberg, ISBN 978-3-642-30745-4, 248–254, doi: 10.1007/978-3-642-30746-120, 2012.

[38] A. C. AB, Applications ready to meet your needs, http://www.axis.com/products/video/compatible_applications/index.php, 2013.

[39] P. Runeson, M. Höst, A. Rainer, B. Regnell, Case Study Research in Software Engineering – Guidelines and Examples, Wiley, 2012.

[40] S. M. Easterbrook, J. Singer, M. Storey, D. Damian, Guide to Advanced Empirical Software Engineering, chap. Selecting Empirical Methods for Software Engineering Research, Springer, 285–311, 2007.

[41] A. Dubois, L.-E. Gadde, Systematic combining: an abductive approach to case research, Journal of Business Research 55 (7) (2002) 553–560.

[42] T. C. Lethbridge, S. E. Sim, J. Singer, Studying Software Engineers: Data Collection Techniques for Software Field Studies, Empirical Softw. Engg. 10 (3) (2005) 311–341, ISSN 1382-3256, doi:10.1007/s10664-005-1290-x.

[43] E. Bjarnason, K. Wnuk, B. Regnell, Are you biting off more than you can chew? A case study on casues and effects of overscoping in large-scale software engineering, Information and Software Technology 54 (2012) 1107–1124.

[44] K. M. Eisenhardt, Building Theories from Case Study Research, The Academy of Management Review 14 (1989) 532–550.

[45] I. Hägg, G. Hedlund, "Case studies" in Social Science Research, European Institute for Advanced Studies in Management 78 (16).

[46] B. Bergman, B. Klefsjö, Quality: From Customer Needs to Customer Satisfaction, Studentlitteratur, ISBN 9789144463315, URL http://books.google.se/books?id=U2avHAAACAAJ, 1994.

[47] O. E. Williamson, Comparative Economic Organization: The Analysis of Discrete Structural Alternatives, Administrative Science Quarterly 36 (2) (1991) 269–296, ISSN 00018392, doi:10.2307/2393356.

[48] T. K. Das, B.-S. Teng, Trust, Control, and Risk in Strategic Alliances: An Integrated Framework, Organization Studies 22 (2) (2001) 251–283, doi:10.1177/0170840601222004.

[49] J. Hagel, J. S. Brown, L. Davison, Shaping strategy in a world of constant disruption, Harvard Business Review (10).

[50] R. Consulting, Risk Assessment Template for Software Development or Acquisition Projects, http://www.niwotridge.com/PDFs/RiskAssesmentTemplate.PDF, 2013.

[51] P. Clements, L. Northrop, Software Product Lines: Practices and Patterns, Addison-Wesley, 2002.

Appendix A. Interview instrument

The purpose of this interview is to obtain a better understanding of how Axis is performing as a platform leader. The interview questions will be focusing on your motivations for (not) developing towards Axis application platform, how Axis? activities as platform leader are perceived and what activities you value.

1. Introduction (10 min)
   (a) Explain that the material is not going to be shared with Axis without consent.
   (b) Explain the purpose of the study.
   (c) One of our goals is to understand what drives development towards a platform. Another goal is to better understand what activities are important to perform as a platform leader to create a good platform and ecosystem.
   (d) Open ended questions where we will restrict you if we have to due to time constraints. Present number of questions and time left continuously.
2. Personal (5 min)
   (a) What is your role at the company today? Previous roles? (years per role)
   (b) (If applicable) How much experience do you have from developing ACAP applications and analytics applications in general?
   (c) In what roles have you acquired this experience?

   (d) Would you call yourself a domain novice or expert within the area of video analytics? On a scale between 1 and 5 where 5 corresponds to being an expert.

3. Characterization of company (10 minutes)
   (a) Tells us about your company! (number of employees, number of products, types of products)
   (b) Tell us about your company's products (number of products, number of products to the ACAP, number of products related to Axis)
   (c) Is your product available as an ACAP application?
   (d) (If applicable) Is your ACAP product also available as a server solution?
4. Reasons for developing (min 30 minutes)
   (a) Why you /your company did chose (not) to start developing an ACAP application?
   (b) (If applicable) How did you come up with the application idea?
   (c) What were the most important factors when choosing the Axis application platform? (Follow leads and dig deeper)
   (d) Was there anything that made you hesitate about developing for the Axis application platform? (Follow leads and dig deeper)
   (e) (If company has chosen not to develop towards the ACAP) What were the most important factors when choosing not to develop towards the ACAP?
   (f) What are the main areas of improvement for ACAP?

Table 2: Interviews, their affiliation, roles, and experience

Company	ADP level	ID	# employees	Role of interviewee	Experience of interviewee	ACAP app.
A	ADP	A1	20	Vice president of research and development (R&D)	5 years as vice president of R&D	Yes
B	ADP	B1	9	International marketing and sales	2+ years at current role. Previous system integrator	Yes
		B2	9	Chief Technology Officer (CTO)	PhD in image processing	
C	ADS	C1	6	Founder and Chief Executive Officer (CEO)	-	No
D	ADP	D1	325	Head of video analytics development group	1,5 years at current position. 10 year within video analytics development	Yes
		D2	325	Director of video management and analytics group	24 years at this company in different roles	
E	ADP	E1	50	Co-founder and CTO	Previously vice president of R&D	No
F	Gold ADP	F1	95	Sales engineer	5 years as a sales engineer and product manager	Yes

Table 3: Summary of the analysis results.

			Bridges		**Barrires**	
Company	Has an ACAP product?	Interviewee	Reasons	Benefits	Hesitations	Drawbacks
A	YES	A1	End customer demand (Re1), Relationship (Re2), Open environment (Re3)	Internal standard (Be1), Piggyback on Axis (Be3)	None	Performance (Dr1)
B	YES	B1	Relationship (Re2), Geography (Re4)	Internal standard (Be1)	None	Performance (Dr1)
		B2	End customer demand (Re1)	Ease of installation (Be4)	None	N/A
C	NO	C1	N/A	N/A	Performance (Dr1), Technical features (He2), Axis' business model (He1), Unclear roles (He3)	Performance (Dr1)
D	YES	D1	End customer demand (Re1), External standard (Be2)	External standard (Be2)	Performance (Dr1)	Performance (Dr1), Debugging (Dr2)
		D2	Future possibilities (Re5), Marketing (Re6)	Internal standard (Be1), Ease of installation (Be4), Less infrastructure (Be5), External standard (Be2)	N/A	Performance (Dr1)
E	NO	E1	N/A	External standard (Be2)	Business model (He1), Lack of information about customers (He4), Strict rules of ADP (He5)	Information-gap (Dr3)
F	YES	F1	Relationship (Re2), Internal standard (Be1), Low risk (Re7)	Scalability (Be6), Less resources needed (Be7)	Performance (Dr1), Maintenance (He6)	Performance (Dr1), No uniform camera compatibility (Dr4)