Software Ecosystem Governance and Participation

A Case Study at Axis Communications AB

Matilda Lantz

Oskar Weijden

Matilda Lantz & Oskar Weijden (aka the Moskar) ©

Department of Computer Science
Faculty of Engineering, Lund University
Ole Römers väg 3
223 63 Lund
Sweden

Department of Business Administration
School of Economics and Management
Tycho Brahes väg 1
223 07 Lund

Master's Thesis Technology Management - Nr 255/2013
ISSN 1651-0100
ISRN LUTVDG/TVTM--13/5255--/SE

E-husets tryckeri, Lund 2013
Printed in Sweden

Abstract

Title: Software Ecosystem Governance and Participation - a Case Study at Axis Communications AB

Authors: Matilda Lantz and Oskar Weijden

Supervisors: Krzysztof Wnuk - *Department of Computer Science*

Carl-Henric Nilsson - *Department of Business and Administration*
Anders Johansson - *Product Manager API & Components, Capabilities at Axis Communications AB*
Johanna Fjelner - *Senior Engineer, Firmware Interfaces at Axis Communications AB*

Issue of Study: Traditionally software development was performed by one single company. However, the traditional roles are being redefined and software ecosystems are taking software development outside the borders of the software company. Companies are starting to collaborate by using digital platforms and are creating new complex networks of organizations, so called software ecosystems. Software ecosystems are a relatively new area of research and a collective theoretical foundation is starting to emerge. In a software ecosystem, surrounding a platform, the governance activities and decisions of the platform leader influence everyone involved. Therefore it is interesting to understanding how the factors behind participation are affected by the governance performed by a platform leader.

Purpose: The purpose of this master's thesis is to explore how governance performed by a platform leader affects developer participation in a software ecosystem. The following three research questions were answered:

1. Why do third party developers to join and participate in Axis' software ecosystem?
2. What makes third party developers hesitant towards joining Axis' software ecosystem?
3. What governance activities are performed by Axis as a platform leader?

Method: Due to the exploratory nature of the master's thesis a case study approach was chosen. The process of work was conducted in accordance with the case study process presented by Runeson and Höst (2009). Qualitative data was collected through interviews at Axis and with external developers connected to the software ecosystem. Additional information was collected through two surveys and written documentation. Through classifying the software ecosystem according to existing classification models and by analyzing the collected data the research questions were answered and created an understanding for how the governance activities performed by the platform leader affect participation in this software ecosystem.

Conclusions: The findings of this master's thesis indicate that both non performed and performed governance activities by the platform leader has an effect on participation in a software ecosystem, and that the contextual factors set the stage for which governance activities that will be most influential. Furthermore, unsystematically performed governance activities were found to increase the need of a personal relationship and good communication between the platform leader and third party developers in order for the latter to participate. Additionally, the creation and stabilization of application programming interfaces (API:s) were found to be important for developer participation in a software ecosystem where the platform leader offer several product lines. Finally, by not selling directly to end customers the platform leader's power is diffused over many vendors. This reduces the effects of governance activities and prevents enactment of others; hence reduce the platform leader's influence on participation.

Keywords: Software Ecosystem, governance, developer participation, platform.

Preface

This master's thesis is the final part of our master's degree in Technology Management. Five years of studies has come to an end and the world is (finally) opening up. Writing this thesis has been a fun and interesting introduction to research and a good closure of our current academic merits. We have come in contact with brilliant people, interesting theoretical areas and we have gotten a glimpse into the world of research. Not only have we learned a lot about the subject at matter but we have also learned a lot about ourselves. For the first time during five years we had the opportunity to focus on one single task for four months, which has been both educational and challenging.

It has been equally interesting and fun to get to know Axis as a company and the people working there. We would like to thank Anders Johansson, Anders Görtz and Johanna Fjelner for the time you spent assisting us. We would also like to thank everyone at Axis who has shared their knowledge and expertise with two, at times very confused, students.

Furthermore we would like to thank Carl-Henric Nilsson at the department of business and administration for his support and eye opening reflections. Last but not least we would like to send a special thank you to Krzysztof Wnuk at the department of computer science. Thank you for all your help, guidance and meticulous proof reading!

Matilda Lantz, Lund, 15th of May 2013

Matilda Lantz

Oskar Weijden, Lund, 15th of May 2013

1 Table of Contents

Definitions

ACAP Developers

ACAP developers are third party developers providing applications to AXIS camera application platform (ACAP, see description below). The developers are a part of Axis' Application Developer Partner (ADP) program in order to get access to the necessary tools to develop.

AXIS camera application platform (ACAP)

AXIS camera application platform is an open application platform that enables development of third party applications. These applications can be downloaded and installed on Axis cameras and video encoders.

Application programming interface (API)

Is a language and message format, a set of rules, used by applications and programs to communicate with the underlying operating system or control program (PCMAG.com, 2012), visualized in Figure 1.

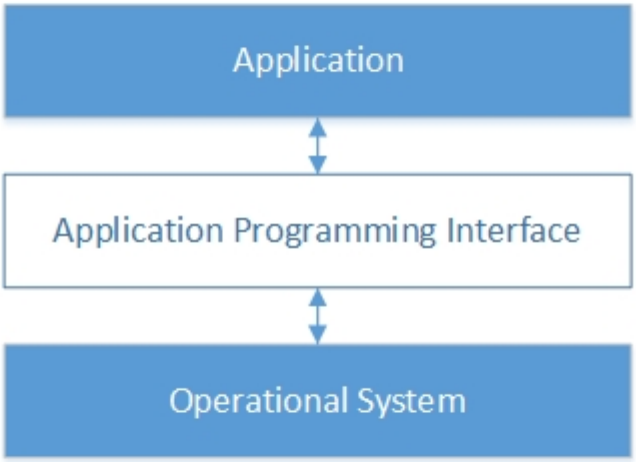

Figure 1 Visualization of an application programming interface

Embedded software

Applications installed and running on the device it is embedded in, as opposed to using features of the device but being installed and ran on an external device.

Governance

Corporate governance

The way a company is managed at the highest level (Lexicon, 2013).

Software ecosystem governance

Procedures and processes by which a company controls, changes and maintains its current and future position in a software ecosystem on all different scope levels (Baars and Jansen, 2012).

Governance activities
Activities performed in order to conduct governance.

Software Ecosystem (SECO)
A set of actors, for example companies, organizations or individuals, functioning as a unit and interacting with a shared market for software and services, together with the relationships among them. These relationships are frequently underpinned by a common technological platform or market and operate through the exchange of information, resources and artifacts. (Jansen et al., 2009)

Third Party Developer
The term third party developer refers to a software or hardware developer who is independent of the primary product or platform.

Video Management System (VMS)
A video surveillance solution consists of cameras, network infrastructure, a storage solution and software for managing the system. This software, called a video management system, is used to managing the system for example to view live or recorded video.

Platform

A hardware or software architecture that serves as a foundation or base.

Platform leader

An organization, actor or community that is responsible for the development and wellbeing of the software ecosystem (Manikas and Hansen, 2013).

Acronyms

A&E:	Architecture and engineering
ACAP:	AXIS Camera Application Platform, the platform underlying the software ecosystem of this study.
ADP:	Application Development Partner
ADP-program:	Axis’ partner program for application developers.
ADS:	Application development service, the lowest level in the ADP-program.
API:	Application programming interface
Axis:	Axis Communications AB
Gold ADP:	Gold Application Development Partner
ID:	Identification
OEM:	Original Equipment Manufacturer
PTZ:	Pan-Tilt-Zoom
R&D:	Research and Development
RQ:	Research Question
SDK:	Software Development Kit
SECO:	Software Ecosystem
SI:	System Integrator
VMS:	Video Management System

2 Introduction

In this section the reader is first given a background to the emerging trend of software ecosystems (SECO) after which the problem statement, purpose and research questions are revealed. Finally the limitations of this thesis are presented.

2.1 Background

In today's fast changing business environment with rapidly changing technology (Viljainen and Kauppinen, 2011) it is crucial for companies to be aware of their competitive capabilities, but also to continuously investigate new ways of competing (D'aveni, 1994). One reason for this is that the customer's willingness to pay for increased technological performance is diminishing (Adner, 2004). This means that when markets are maturing and companies are continuing to develop their technology, the customer's basic needs are increasingly becoming fulfilled. Hence technology leaders are less able to charge a premium for the added performance of their technology and thus less able to leverage their past competitive advantage.

A way to stay competitive is to change the rules of the game (D'aveni, 1994). Kim and Mauborgne (1997) found that companies in these situations sometimes attempt to rejuvenate the market and alter the customer's perception of so-called performance attributes. One way of changing the competitive focus is to use software to add additional value. Boncheck and Paul (2013) presented the importance of software platforms in order to create additional value. They introduced Nike as an example and their success in digital sports products. Nike recently launched Nike+ Accelerator a tool enabling companies to build on their Nike+ platform. In construction, a platform is something that lifts you up and on which others can stand. The same is true in software. By building a digital platform, other businesses can connect their business with yours, build products and services on top of it, and co-create value (Bonchek and Paul Choudary, 2013).

Traditionally software development was performed by one single company. However the traditional roles are being redefined and according to Manikas and Hansen (2013) software ecosystems are taking software development outside the borders of the software company. Additional value is created and new business models are emerging when companies are starting to compete on new levels (Manikas and Hansen, 2013). Modern software rather relies on components and infrastructure from third party vendors or suppliers than on one single company (Cusumano, 2004). Relationships between these actors shape the product software landscape into a software ecosystem where they collaboratively create competitive value (Jansen and Cusumano, 2012).

"...the success of a product software company therefore no longer depends only on its own development quality but also on the way it manages its relationships"

Jansen and Cusumano (2012)

2.2 Problem Statement

The area of Software ecosystems is a relatively new field of research and a collective theoretical foundation is starting to emerge. In a software ecosystem, surrounding a platform, governance activities and decisions of the platform leader influence everyone involved (Gawer, 2007). By understanding the factors behind participation of their software ecosystem, platform leaders will be better equipped to facilitate a successful software ecosystem (Gawer, 2007). Additionally, a company benefits from understanding what consequences their governance activities have (Clarke and dela Rama, 2008). Some research, such as Baars and Jansen (2012), focus on software ecosystem governance but they call for more case studies to formalize this area of research.

"Now that a list of governance tools has been created, the applicability of these tools and the dependent situational factors must be determined".

Jansen and Cusumano (2012)

Historically studies have mainly been conducted on software companies. However, Axis Communications AB (Axis) is a company producing video surveillance cameras, i.e. focused on hardware, which is now showing an increasing interest in their software ecosystems. Therefore the authors see an opportunity to explore previous research in a new context of hardware dependent and young software ecosystem. How does Axis' context and hardware focus connect to previous research? How is Axis affecting its SECO and what are the driving factors behind participation in this type of software ecosystem?

2.3 Purpose

Explore how governance performed by a platform leader affects developer participation in a software ecosystem.

2.3.1 Research Questions

The purpose will be reached by answering the following three research questions based on a case study performed at Axis Communications AB (Axis).

1. Why do third party developers join and participate in Axis' software ecosystem?
2. What makes third party developers hesitant towards joining Axis' software ecosystem?
3. What governance activities are performed by Axis as a platform leader?

2.4 Scope and Limitations

This master's thesis is focused on the context surrounding Axis due to them showing an interest in widening their business through software ecosystems. AXIS camera application platform (ACAP) is chosen as the main focal point of this SECO as it is expressed to be an interesting area to expand for Axis.

Due to time and availability constraints, this master's thesis investigate the developer side of the platform and does not further look into the sales channels or end customers. For the same reason focus is held on a company and strategic level. Detailed architectural problems, for example API design and documentation are disregarded.

3 Methodology

This section presents the choice of research methodology, work process and finally a discussion about validity.

3.1 Starting Point

This master thesis was conducted as a part of the master's degree Technology Management at Lund University. One of the authors originally studied at the School of Economics and Management in Lund while the other studied engineering physics at the Faculty of Engineering. The master's thesis was initiated by Anders Johansson, product owner of AXIS camera application platform at Axis Communications AB.

Initially the authors were provided with an area of research, AXIS Camera Application Platform, with additional focus on software developers surrounding the platform. These areas were chosen because Axis was interested in increasing their understanding of how they could improve their software ecosystem and increase the number of developers connected to the platform. The product owner of ACAP at Axis was appointed as supervisor of the master thesis and an initial meeting was conducted where Axis declared their awareness of the broad scope of their proposal and gave the authors time and opportunity to come up with research questions of a smaller scope within the same area.

3.2 Choice of Research Methodology

The purpose of a study may vary, ranging from descriptive, exploring, explanatory or problem solving (Höst, 2006). Due to Axis' wishes and the field of software ecosystem being relatively new, the aim of this thesis was to be exploratory in a new context, see purpose in section 2.3. Exploratory studies are often used when looking for deeper knowledge and understanding and is most suitable when researchers want to describe as well as explain a phenomenon (Björklund, 2003). The choice of study is highly dependent on the purpose of the thesis as well as the current body of knowledge in the field studied (Björklund, 2003). Historically case studies have often been used for exploratory purposes (Runeson and Höst, 2009) and therefore deemed appropriate in the case of this thesis.

"A case study examines a phenomenon in its natural setting, employing multiple methods of data collection to gather information from one or a few entities (people, groups, or organizations) "

Benbasat et al (1987)

Performing multiple case studies of different SECO:s, instead of a single one, was not possible due to lack of data accessibility. An alternative would have been to conduct a survey targeting developers connected to ACAP. The risk of low respondent rates and the impact that would have on the result was however seen as too large. The choice of a case study is further supported by Runeson and Höst (2009) who stated that a case study is a suitable research methodology for software engineering research since it studies contemporary phenomena in its natural context.

3.3 Work Process

This thesis followed the case study process proposed by Runeson and Höst (2009) which contained five steps:

1. Case study design
2. Preparation for data collection
3. Collecting data
4. Analysis of collected data
5. Reporting

The authors added a step, (0) pre-study, in order to gain initial information about the research area of software ecosystems. However a case study methodology is a flexible design strategy (Andersson and Runeson, 2007) and iterations have been done between the steps. The work process and steps of this thesis is visualized in Figure 2 and described in more detail in section 3.3.1–3.3.7 below.

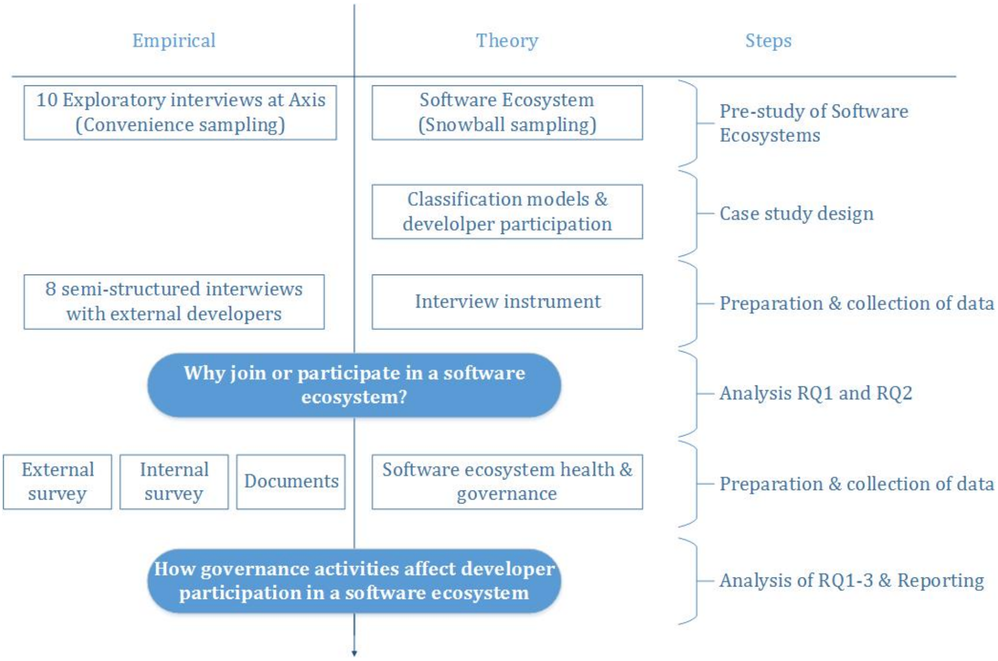
Empirical
Theory
Steps
10 Exploratory interviews at Axis (Convenience sampling)
Software Ecosystem (Snowball sampling)
Pre-study of Software Ecosystems
Classification models & develolper participation
Case study design
8 semi-structured interwiews with external developers
Interview instrument
Preparation & collection of data
Why join or participate in a software ecosystem?
Analysis RQ1 and RQ2
External survey
Internal survey
Documents
Software ecosystem health & governance
Preparation & collection of data
How governance activities affect developer participation in a software ecosystem
Analysis of RQ1-3 & Reporting

Figure 2 A visualization of the iterative work process where empirical and theoretical studies are performed in parallel. The connection to Runeson and Höst's (2009) proposed steps are presented to the right in the figure.

3.3.1 Pre-Study

Initial material on software ecosystems were provided by the supervisors. A structured literature study was deemed too time consuming and instead new material was identified through interesting references in the initial ones. This type of sampling strategy is commonly called snowball sampling.

With an initial understanding of software ecosystems the authors started an iterative process between studying theory and collecting empirical data. Through continuous iteration between the empirical and theoretical world the authors are able to expand their understanding of both theoretical and empirical phenomena (Dubois and Gadde, 2002). 10 exploratory interviews were conducted with employees at Axis (see Table 6 in Appendix A). The interviewees were identified through recommendation by the authors' supervisor at Axis and by recommendations from the initial interviewees, a procedure referred to as convenience sampling (Parasuraman, 2004). Participants possessed relevant experience of Axis, ACAP and/or software development. By performing convenience sampling some people may have been disregarded and some information might have been overseen. That risk was reduced by studying additional material such as documents, reports and relevant information at Axis' intranet.

The pre-study gave the authors the basic knowledge and understanding of both software ecosystem theory and the situation at Axis. Thereafter interesting research areas could be identified for further investigation.

3.3.2 Case Study Design

This step includes planning the case study, setting the objectives as well as the supporting research questions (if any are used) (Runeson and Höst, 2009). As mentioned in section 2.3 above, the objective and purpose of this thesis is to "Explore how governance affects developer participation in a software ecosystem". This purpose is attained by answering three research questions (see section 2.3.1). Research questions one and two (RQ 1 and 2) focus on developer participation and research question three (RQ3) focus on the performed governance activities. The scope of the case study was first defined as Axis together with the group of companies currently developing towards ACAP. Companies that have chosen not to develop toward ACAP was also identified as relevant and hence included in the case.

After the initial pre-study, material on software ecosystem classification and related theory on developer participation were studied. These theories formed the foundation of the study and were primary used to provide a frame of reference of the study. They make the context of the case study clearer and provide help both for the authors and people who review the thesis (Runeson and Höst, 2009).

Case studies often include several different methods of collecting data such as interviews, observations and surveys. This data can either be quantitative or qualitative (Eisenhardt, 1989). The Interviews were used to collect primary data for research question one and two, further explained in section 3.3.3.1. Additional empirical data were collected through two surveys, meetings, documents and observations at the case company. Focus was on qualitative methods as they were deemed most suitable due to qualitative data providing a richer and deeper description (Björklund, 2003).

3.3.3 Preparation and Collection of Data for RQ1 and RQ2

The empirical data for research question one and two were collected through eight semi-structured interviews and a survey. Interviews are classified as a direct data collection technique (Lethbridge et al., 2005) 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 (Runeson and Höst, 2009).

The interview questions were based on information obtained during the pre-study presented in section 4.4.1. Semi structured interviews were seen as most appropriate due to the goal of the thesis being exploratory. Semi structured interviews can help to ensure that common information between interviewees is collected and allow the interviewer to dig deeper when required (Bjarnason, 2012). All planned interview questions were included in an interview instrument (presented in Appendix B) which was used as a template during the interviews. The interview instrument was reviewed by one of the supervisors (department of computer science) who has conducted similar studies and can be viewed as experienced in that area. Preparations were made before each interview in order to adapt the questions to characteristics of that specific interviewee and company.

The interviews were conducted through loudspeaker phones, except in one case where the interview was conducted in person. A phone was used due to a wide geographical spread of the interviewees. All interviews except one were performed in English. Two interviewers were present at all interviews. This enabled 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.

3.3.3.1 Choice of Interviewees

A number of criteria were set up to make the choice of interviewees. The criteria were set up on a company and individual level. On a company level the first criterion was that they had an ACAP application available. This was set as the criteria for participation. The second criterion was on an individual level, the individual had to have been involved in the decision process to develop towards ACAP. The third criterion was that the individual had to have both technical and business expertise. If it was not possible to find an individual who fulfilled both of these requirements, an option of interviewing two individuals was deemed the next best choice. The fourth criterion was that of availability, they had to have the spare time for an interview of one hour, a follow up survey and follow up questions.

Companies developing towards ACAP were available through the list of available ACAP applications on Axis website, in total 16 companies. There were 4 companies that fulfilled the above mentioned criteria, as the companies were small and relatively new, a person who had been involved in the decision process most often had both business and technological expertise. In two cases two interviewees were needed. Availability of a person who had been involved in the decision process had the largest impact on the sample size. Two companies that were not developing towards ACAP, but had shown an interest towards the platform, were also interviewed. The second, third and fourth criteria presented above were applied to them as well. They were however not systematically chosen, but suggested by Axis. Both companies had previously expressed an interest in developing towards ACAP but had for different reasons chosen not to complete the development of an ACAP application. Their expressed interest indicates that they at one point saw potential in the platform, which could affect their answers. However, in order to answer relevant questions related to the platform the company had to possess information about it.

3.3.4 Data Analysis Techniques of RQ1 and RQ2

Analyzing data is the heart of building theory from case studies, and it is both the most difficult and the least codified part of the process (Eisenhardt, 1989). The recorded and transcribed interviews were gone through and answers connected to RQ1 and RQ2 were given ID:s. A structured approach is important when conducting a qualitative analysis and hence the authors used tabulation where the coded data was arranged into tables (Table 2, 3 and 4 in Appendix A), a method suggested by Runeson and Höst (2009). This method provided an overview of the collected data.

According to Eisenhardt (1989) 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 done in order to identify patterns or connections, a technique that is called explanation building (Runeson et al., 2012). Finally a cause and effect analysis of the identified reasons was performed in order to reveal underlying factors affecting the findings. The cause and effect analysis was based on the theory connected to the classification models (see section 4.3.1 and4.4). The cause and effect analysis was originally included in the interview questions. But the information collected during the internal and external interviews, as well as information obtained from documents, were connected to the theory and used throughout the analysis.

3.3.5 Preparation and Collection of Data RQ3

The authors investigated previous related work on software ecosystem, business ecosystem and governance in order to obtain a starting point in already identified relevant governance activities. As a complementary measure the factors identified as affecting participation from the cause and effect analysis of RQ1 and RQ2 was also included if assessed relevant. This was then complemented with relevant theory in order to expand and explain why each activity should be performed and what possible effects they could have.

Following the semi-structured interviews described in section 3.3.3 a survey was sent out to the interviewees. A reason for performing a survey instead of additional interviews was time constraint and availability of the interviewees. The questions included in the survey were connected to the governance activities in order to investigate if and how the interviewees perceived them to be conducted. After some moderations the survey was also sent out internally at Axis in order to identify possible discrepancies between the developers and Axis' employees. It was sent out to 25 employees at Axis with relevant knowledge or connections to ACAP (presented in Appendix C). However the surveys were not filled out in an adequate way and hence did not contribute to the analysis as expected. It did however validate answers that were received during the interviews both with developers and personnel at Axis. Additional information about Axis' activities was gathered through studying written material at Axis website and intranet as well as informal meetings with employees at Axis.

3.3.6 Data Analysis Techniques RQ1, 2 and 3

An analysis was performed on how the governance activities performed, or not performed, affected the findings of RQ1 and RQ2, participation in large as well as the climate of the software ecosystem. Due to the choice of research design being case study, contextual factors were deemed to have a significant effect on participation and were hence also included in the analysis. By identifying connections and correlations between governance, or lack thereof, contextual factors and the identified reason and hesitations to participate, the authors created an understanding of how governance affect the participation in Axis' ACAP SECO.

3.3.7 Reporting

The result of this master's thesis is reported through this very document which is printed and distributed among supervisors as well as published on the website of the master program Technology Management. Two oral presentations were performed: one at the case company Axis and one at Lund University. In accordance with set up rules a one page article of the master's thesis was also constructed.

3.4 Validity

Validity is defined as to what extend researchers measure what was intended to be measured (Björklund, 2003). Runeson and Höst (2009) distinguish between four types of validity: construct validity, internal validity, external validity and reliability.

Construct validity refers to what extend the operational measures that are studied represent what the researchers planned to investigate. A threat towards construct validity is if for example interview questions are not interpreted in the same way by the researchers and the persons interviewed (Runeson and Höst, 2009). In order to minimize this risk the authors of this thesis tested their interview and survey questions on three employees at Axis as well as their supervisors in two iterations before presenting them to the interviewees.

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 (Runeson and Höst, 2009). Members of a software ecosystem are often described as closely affecting each other in complex networks (Bosch, 2009). This threat towards internal validity is important to be aware of and is reduced by extensive classification in order to map possible factors and how they affect each other. It was also reduced by collecting data from several sources, internal at Axis and from external developers. This method of taking different angles towards a studied object is called triangulation and it is important when conducting a qualitative study (Runeson and Höst, 2009).

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 (Runeson and Höst, 2009). There are researchers that mean that case studies cannot be generalized (Hägg and Hedlund, 1978) while others mean that a generalization can be made as long as an extensive characterization is performed along with analysis, which captures phenomena that the researcher think can be applied to other companies or areas of research (Norrman, 1980). Runeson and Höst (2009) suggested 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. In order to facilitate generalization and comparison with other cases classifications according to existing classifications models were performed on the case company of this thesis.

Reliability refers to the aspects of to what extent the data and the analysis are dependent on the researchers (Runeson and Höst, 2009). 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. Both authors were present at all interviews in order to reduce bias.

4 Theory and Related Research

This section presents theory and related research which provide a frame of reference and understanding relevant to the purpose of the master's thesis. The use of common concepts and tools allows for contextual comparisons. A background to the concept of software ecosystems is first presented. Followed by different types of actors within a SECO and how a SECO can be classified. Finally, the concept of SECO health is introduced followed by SECO governance and relevant governance activities available to a platform leader.

4.1 Biological Ecosystems

The term business ecosystem originates from analogies made with biological ecosystem first presented by Moore (1993). Iansiti and Levien (2002, 2004b, 2004a) and Dhungana et al (2010) further developed these analogies to establish the relationship between biological ecosystems and the appearance and functionality of business respectively software ecosystems. These analogies can however only be stretched so far. Jansen and Cusumano (2012) presented two main differences between the two. Firstly, biological ecosystems are mainly studied to observe influences from external factors, whereas the dynamics of ecosystems within the software industry are analyzed mainly with the aim of growth and success. Secondly, the participants in biological ecosystems do not consciously take part while software ecosystem participants can both consciously leave the ecosystem as well as destroy it. Nevertheless, the frameworks developed with its base in biological ecosystems provide a useful terminology when it comes to analyzing business and software ecosystems.

4.2 Business Ecosystems

Moore (1993) proposed the view of looking at a company not as a member of a single industry but as a part of an ecosystem. He defines business ecosystems as an economic community supported by a foundation of interacting organizations and individuals, additionally including suppliers, lead producers, competitors, and other stakeholders. According to Moore (1997) members of a business ecosystem over time co-evolve their capabilities and roles and tend to align themselves with the central company of the ecosystem. Software ecosystems are seen as a subset of business ecosystems (Jansen and Cusumano, 2012), which opens up the use of business ecosystem theory to be applied on software ecosystems.

4.3 Software Ecosystems

The definition of a software ecosystem that will be used throughout this thesis is found to be the most commonly used by Manikas and Hansen (2013). It was introduced by Jansen and Cusumano (2012) based on earlier work by Kittlaus and Clough (2009) and Bosch (2009):

"A software ecosystem is a set of actors functioning as a unit and interacting with a shared market for software and services, together with the relationships among them. These relationships are frequently underpinned by a common technological platform or market and operate through the exchange of information, resources and artifacts."

Definition of software ecosystems by Jansen and Cusumano (2012)

An interesting question is what benefits a software ecosystem potentially could yield. Barbosa and Alves (2011) performed a systematic mapping study where they compiled current studies on identified benefits of SECO:s. According to their results the most mentioned benefits of a software ecosystem were fostering of co-evolution, innovation and increased attractiveness for new players as well as collaboration and knowledge-sharing. Fostering of innovation and collaboration are also presented by Bosch (2009) as potential benefits.

4.3.1 Types of Actors within Software Ecosystems

According to the definition presented above, a software ecosystem is constituted by a set of actors. Manikas and Hansen (2013) performed a systematic literature review of software ecosystems where they compiled the most common actors encountered in currently produced literature. These four actors are presented below together with corresponding concepts.

4.3.1.1 Platform Leader

Platform leader is a concept presented by Gawer (2002), corresponding to keystone player (Hanssen and Dybå, 2012, Iansiti and Levien, 2004b, Iansiti and Levien, 2004a), orchestrator (Runeson and Höst, 2009) and coordinator (Jansen and Cusumano, 2012). This is an organization, actor or community that is responsible for the development and wellbeing of the software ecosystem (Manikas and Hansen, 2013). The position of platform leader is a powerful position (Iansiti and Levien, 2004b). According to Manikas and Hansen (2013) this unit typically governs the software ecosystem by for example running the platform, creating rules and monitor quality standards. Platform leaders enable members to move toward shared visions, aligned investments and mutual support (Gawer and Cusumano, 2002). The software ecosystem can either be owned by one private entity or multiple, making it a community (Jansen and Cusumano, 2012).

4.3.1.2 Vendors

Vendors, including resellers and value-added resellers, are businesses that profit from selling products created by the software ecosystems. They sell to customers, end users, or other vendors and sell for example complete installations, components, leasing contracts or support agreements. (Manikas and Hansen, 2013)

4.3.1.3 End Customers

End customers or end users purchase complete or partial solutions from actors within a software ecosystem. They need the ecosystems in order to carry out their business (Manikas and Hansen, 2013).

4.3.1.4 External Actors and Niche Players

Manikas and Hansen (2013) identify a fourth role called external actors, which is used to describe third party or external developers that develop on top of the software ecosystem platform. An external actor is similar in its description to a niche player identified by Jansen et al (2009). A niche player is an actor that requires the technological standard or platform provided by a platform leader to create business value (Jansen et al., 2009). The difference described by Manikas and Hansen (2013) is that a niche player develops components to the platform and provides functionality that the customers require, adding direct value to the software ecosystem. An external actor on the other hand provides indirect value from its use of the platform.

4.4 Classifications of Software Ecosystems

The definition of software ecosystems presented in section 4.3 is quite broad and thus fits a wide array of software ecosystems. To differentiate between different types of software ecosystems Jansen and Cusumano (2012) developed a classification model. By applying an already developed and used classification model on Axis comparisons with other software ecosystems are more accessible and thus situational factors more identifiable. The classification model is based around four characterizing factors: (1) base technology, (2) accessibility, (3) extension market and (4) platform leader. Jansen and Cusumano (2012) additionally mentioned networking effects as interesting characterization criteria but chose not to include it due to that it cannot be assessed binary i.e. by a yes or no. Networking effects are however closely connected to software ecosystems (Iansiti and Levien, 2004a). Kato and Negoro (2007) also reported that networks effects were one of the key mechanisms of a successful platform product. It is therefore included in this study which has no requirement of binary assessment.

4.4.1 Base Technology

In Jansen's and Cusumano's (2012) research they reported that all software ecosystems are underpinned by a technology. They found that the underpinning technology can be split in three categories: platform, service platform and standard. The first category is the most common type where the platform can be installed on a physical product. The service product is different from the platform in the sense that it is only available online. A taxonomy presented by Bosch (2009) was found to be a good complement to the original classification of base technology. Bosch (2009) reported on three different types of software ecosystems: operating-centric, application-centric and end user-centric software ecosystems.

Operating system-centric software ecosystem has a software platform where the underlying hardware only is implied. For example, when an application is said to run on the Windows platform it means that the program has been compiled into the x86 machine language and runs under the operative system Windows (PCMAG.com, 2012). Bosch (2009) presented four main characteristics for this type of software ecosystem. First, these kinds of software ecosystems are domain independent and assume third party developers to build applications that offer value to customers. Second, the platform has to be present at all devices using it. Hence, the ecosystem becomes affected by sales of these devices. Third, this type of software platform typically optimizes development for standalone applications and offer little support for cross-application integration. Finally, an operating system-centric software ecosystem offers development tools to simplify usage and adoption for developers.

Application-centric software ecosystem is organized around an application (Bosch, 2009). It is often domain specific and is usually originating from an application first being successful without a surrounding ecosystem. When additional requests are made that the company cannot fulfill those are usually solved by creating API:s. When opening up the application it turns into a domain specific platform accessible for third party extensions. According to Bosch (2009) application-centric software ecosystems are characterized by the platform provider offering techniques of making third party applications seamless to end customers. Third party developers tend to extend the functionality offered by the platform leader.

End user programming-centric software ecosystems most commonly take the form of a domain specific programming language (Bosch, 2009). It enables end-users with no computer science or engineering degree to create applications on their own, for example MS Excel (Bosch, 2009). The type of platform required to create this type of software ecosystem is according to Bosch (2009) considered a holy grail. The creation is however very difficult, the amount of work required to create a programming language tends to limit the scope of it and thus create a very domain specific software ecosystem (Bosch, 2009).

4.4.2 Accessibility

A factor which characterize a software ecosystem is how accessible it is and what barriers of entry it has (Jansen and Cusumano, 2012). This accessibility determines what actors that join and how they act. A software ecosystem is either open source, screened but free or paid (Jansen and Cusumano, 2012). These characteristics are however not entirely mutually exclusive as both open source and paid could be both screened and unscreened. The highest level of restriction would be both paid and screened.

4.4.3 Extension Market

Jansen and Cusumano (2012) described that many software ecosystems are centralized around a marketplace for extensions to the software ecosystem, which is for example third party applications. There are software ecosystems with many such marketplaces and others with none. They can also range from comprehensive marketplaces such as the well-known Apple App-store, enabling both distribution and sales, to a smaller simple list only providing information of availability.

4.4.4 Network Effects

The basic economic term of network effects is commonly used in literature about economics of platforms. Liebowitz and Margolis (2001) defines a network effect as an increasing benefit surplus received by an agent due to increased consumption of the same good by other agents.

Authors described different types of network effects such as direct network effects where increased usage lead to direct increased value (Farrell and Klemperer, 2007), with a common example being a telephone. Indirect network effect are when increased usage of a product triggers production of complementary products which in turn results in increased value of the original product (Farrell and Klemperer, 2007). Finally, two-sided network effects are related to two sides of users connected by an in-common platform. Increased usage at one side will create value of complementary products on the other side (Rochet and Tirole, 2004). An example of this is when developers choose to develop code for an operative system because it has many users, and users choose to adopt the same operative system because it has many developers.

4.5 Ecosystem Health as a Measure of Performance

Ecosystem health was first introduced as an overall performance indicator by Iansiti and Levien (2002). The term health was taken as an analogy from biology where it refers to the status of a system or a species. When Iansiti and Levien (2004b) referred to a healthy business ecosystem they meant an ecosystem that was "durably growing opportunities for its members and for those who depend on it". Ecosystem health constitutes three aspects: productivity, robustness, and niche creation which will be further described below.

4.5.1 Productivity as a Health Indicator

The most important health measure of an ecosystem is its productivity, i.e. its ability to transform for example technology into new products or decreased costs (Iansiti and Levien, 2004b). This could also be seen as its ability to create value. Another way of looking at productivity is in the form of resource management, i.e. how well the ecosystem balances the use of scarce resources with best benefits (Dhungana et al., 2010). Iansiti and Levien (2004b) take return on invested capital as an example metric of productivity.

4.5.2 Robustness as a Health Indicator

Derived from network theory, robustness is important for an ecosystem in order to endure disruptions. It includes the ability to deliver value to its participants over time (Iansiti and Levien, 2004b). A robust ecosystem is relatively predictable and relationships between participants enjoy a buffer against external disruptions. Disruptions could for an example come in the form of unforeseen technological change or new competitive forces. Iansiti and Levien (2004b) suggest measuring robustness in the form of survival rate within an ecosystem.

4.5.3 Niche Creation as a Health Indicator

Iansiti and Levien (2004b) reported that robustness and productivity are not enough to capture the health of an ecosystem. They stated that diversity increase an ecosystem's ability to absorb external shocks and its potential to produce innovation. Furthermore they stated that the best measure of this ability is the system's ability to increase *meaningful* diversity, meaning valuable new niches or functions. Dhungana et al (2010) reported that with high diversity in a software ecosystem the risks are spread between different developers and user groups, enabling a software ecosystem to continue if one market segment perishes.

4.6 Software Ecosystem Governance

In order to understand the concept of SECO governance one has to start by understanding governance at large. Governance is a widely used term used in many different contexts without a unified definition. Lamm (2008) described corporate governance and governance activities as "the overall management approach through which senior executives direct and control the entire organization, using a combination of management information and hierarchical management control structures. Governance activities ensure that critical management information reaches the executive team and that it is sufficiently complete, accurate and timely. This enable appropriate decision making, and provide the control mechanisms to ensure that strategies, directions and instructions from management are carried out systematically and effectively" or in more simple terms: the way a company is managed at its highest level (Lexicon, 2013). There is often confusion between the terms governance and the closely related concept of management. According to Van Grembergen (2004) management has a main focus on the internal domain and handle the internal management of present operations. Although governance also includes internal businesses, the same author stated that governance is broader and focus on both the internal domain as well as the external. Governance works with meeting current and future demands of both the business and customers (Van Grembergen, 2004). In the context of this thesis the two concepts can be viewed as overlapping with a wider applicability of governance.

Governance is important to stakeholders of a corporation or a society. It generates a good economic environment by ensuring that the governed entity is acting in line with the stakeholder's expectations. Similarly a healthy software ecosystem has to meet expectations from its participants. The main difference between corporate governance and software ecosystem governance is the central point of focus. For SECO governance the scope includes participants, and their relationships, underpinned by a common technological platform instead of an organization and its stakeholders. Jansen and Baars (2012) have defined software ecosystem governance as:

"Procedures and processes by which a company controls, changes or maintains its current and future position in a software ecosystem on all different scope levels"

Definition of SECO governance by Baars and Jansen (2012)

4.7 Governance Activities in a Software Ecosystem

Baars and Jansen (2012) conducted a literature study and composed a list of observed governance activities. These activities were compiled into a framework which was further developed by Jansen and Cusumano (2012). They combined it with software ecosystem health and conducted 19 case studies on relevant software ecosystems to evaluate the model. This model was used as a starting point and complemented with relevant theory. The result is presented below, divided in to the three categories of SECO health. In section 7 these activities are compared to the activities of the case company and in section 8 their affect upon the software ecosystem are analyzed.

4.7.1 Governance Activities connected to Niche Creation

4.7.1.1 Expand Applicability

The importance of expanded applicability is connected to the value of a platform increasing by association of niche players. Niche players are able to create new products that attracts customers to the platform that otherwise would not have been reached (Jansen and Cusumano, 2012). Ecosystems which contain many diverse niches allow participants to specialize in their areas and avoid head-on competition (Hagel et al., 2008). Such diversity tends to be created from platforms offering a wide set of functionality (Hagel et al., 2008), hence by expanding the applicability of the platform a platform leader can facilitate a larger variation of participants.

4.7.1.2 Make Strategy Explicit

Through making the strategy explicit niche players are assured of their future position within the SECO (Jansen and Cusumano, 2012). Other authors (Viljainen and Kauppinen, 2011, Hagel et al., 2008, Bosch, 2009) described similar actions. Viljainen and Kauppinen (2011) presented it as sharing a vision, Hagel et al (2008) calls it shaping acts and Bosch (2009) referred to a roadmap. Summative it is supposed to create trust among participants towards the platform leader's intention and commitment.

4.7.1.3 Create API

Creating an API standardizes the way that third party developers interact with a platform which decreases compatibility issues providing the platform leader with more control over the platform and how it is used (Jansen and Cusumano, 2012). It will also make it easier for niche players to develop components for the platform, in turn increasing both productivity (van den Berk et al., 2010) and niche creation.

4.7.1.4 Co-development

Co-development could be used to attract niche players to the SECO (Jansen and Cusumano, 2012). A platform leader often possesses detailed knowledge about the platform and niche players have more information about their domain (Hanssen, 2012). By performing co-development synergies can be obtained which can drive innovation, reduce costs and development time (Collabnet, 2004, Chesbrough, 2003).

4.7.1.5 Develop new business models

Platform leaders can enable new niches and business opportunities by introducing new business models to third parties. This could for example be done through introducing a marketplace which enables third party developers to reach customers they would not have reached on their own. (Jansen and Cusumano, 2012). This can be connected to what Hagel et al (2008) reported on what a platform leader should provide the SECO. They suggested that it is the platform leader's responsibility to provide focus through identified business opportunities and forces connected to the SECO.

4.7.2 Governance Activities Connected to Robustness

4.7.2.1 Create Partnership Model

A partnership model allows the platform leader to set up rules and processes to which the partners must adhere (Jansen and Cusumano, 2012, Baars and Jansen, 2012). Tiers could be incorporated into the model and raise incentives for developers by having the higher levels offer increased support and access to more information (Jansen et al., 2011).

4.7.2.2 Do Marketing

Awareness among customers and developers is a prerequisite to be able to assess if the platform is a viable alternative. Hence marketing of the platform is an important activity to increase participation. Jansen and Cusumano (2012) found that the ecosystem can be strengthened through marketing activities by the platform leader.

4.7.2.3 Partner Development Program

Both strong and weak partners can affect the stability and performance of an ecosystem (Jansen and Cusumano, 2012). The weaker less productive companies might lower the overall health of the SECO (Hartigh et al., 2006) for example through signaling low performance to other developers outside the SECO. By a partner development program a platform leader can strengthen these weak actors and hence the entire SECO (Jansen and Cusumano, 2012) and increasing the attractiveness for other participants.

4.7.2.4 Form Alliances

An alliance is defined as an agreement on interaction (Nationalencyklopedin, 2012). By creating alliances with other ecosystems the platform leader can facilitate the creation of complementary sub-groups of participants. This makes a SECO more robust (Jansen and Cusumano, 2012) because in the case of a disruption one sub-group could survive while other reconcile (Iansiti and Levien, 2004a). Furthermore strategic relationships with major incumbents in a market can create an accelerating aggregation of participants due to for example increased awareness and accessibility to their contacts (Hagel et al., 2008).

4.7.2.5 Stabilize API:s

In order to simplify the software configuration and experience of the end customer backward compatibility is important (Viljainen and Kauppinen, 2011). Stable API:s create consistency within the SECO allowing the actors within the SECO to create "trustworthy and stable extensions to the platform" (Jansen and Cusumano, 2012). Unstable API:s could incur developers with increased costs of maintenance of old versions, upgrading and changing to another technology (Hartigh et al., 2006).

4.7.2.6 Raise Entry Barriers

The term entry barriers means an impediment that makes it more difficult for a firm to enter a market (Directorate for financial and enterprise affairs, 2006). There are several different entry barriers into a SECO and knowledge about these entry barriers provides the platform leader with one of the keys to growing the SECO (Jansen et al., 2009). The entry barriers of a SECO are there to ensure that only the right companies can join. If entry barriers are too low, the stability of the SECO might decrease because of uncontrolled growth and loss of quality (in developers or the components they develop) and thereby the risk of an unhealthy ecosystem increases (van den Berk et al., 2010). Jansen and Cusumano (2012) promoted raising entry barriers for example through fees or certifications in order to create higher quality. By creating a certification program for the applications the platform leader is ensured that the applications reach a certain level of performance and hence not damage the SECO by underperforming. Additionally, with more rigorous screening customers and other developers might perceive it as a sign of quality to obtain a membership

(Rao, 1994). However if the barriers become too high they might exclude too many developers and impose a risk to innovation (van den Berk et al., 2010).

Technologies and platforms incorporated into a SECO define what an actor must adopt to become a member (Jansen et al., 2009) and is therefore also acting as an entry barrier.

4.7.2.7 Propagate Operational Knowledge

Software operational knowledge is defined as knowledge of in-the-field-performance, quality and usage, of software and experience feedback (van der Schuur et al., 2011). Through communicating this knowledge to the actors of the SECO the platform leader can improve the quality and their performance towards end customers (van der Schuur et al., 2010), i.e. increase their operational knowledge (Jansen and Cusumano, 2012) and hence increase productivity.

4.7.3 Governance Activities Connected to Productivity

4.7.3.1 Organize Developer Days

Connections are created through interaction between members of the SECO and a good way to facilitate members meeting is to arrange developer days (van den Berk et al., 2010). Several researchers have investigated the importance of connections within a network. Van den Berk et al (2010) reported that platform leaders should try to create a higher degree of connectedness between third party developers in order to increase robustness. An ecosystem with more internal connections is more robust because in case of a disruption the probability of survival increases (Iansiti and Levien, 2004a). Furthermore, more relations with others create more opportunities for the companies (Sydow and Windeler, 1998 cited in Hartigh et al., 2006) and it can also be used as a way of raising awareness of the platform (Jansen and Cusumano, 2012)

4.7.3.2 Collaborative Marketing

Collaborative marketing is conducted when two or more companies join their marketing efforts to create higher customer awareness (Bucklin and Sengupta, 1993). The activities are targeted at the benefits derived from the complementarities of the products (Bucklin and Sengupta, 1993). By pooling resources the companies can gain higher potential reward (Bucklin and Sengupta, 1993). A platform leader is often larger than the third party developers and when it conducts collaborative marketing with the smaller third party developer it emphasize a respectable relationship between the two firms (Jansen and Cusumano, 2012) leveraging the platform leaders incumbent reputation.

4.7.3.3 Create Sales Partner Program and New Sales Channels

Finally, by creating new sales channels and setting up sales partner programs the platform leader can help to create more value by connecting niche players to customers in new fields and enable more revenue for the ecosystem participants (Jansen and Cusumano, 2012). Furthermore, good sales support performed by the platform leader has proven to increase the profit margins of the niche players as well as for the platform leaders themselves (van den Berk et al., 2010).

5 Case Presentation and Classification

This section starts with a presentation of the network video surveillance market followed by a presentation of Axis as a company. Thereafter the SECO surrounding ACAP is classified and described according to the models and concepts presented in section 4.3.1 and 4.4.

5.1 The Network Video Surveillance Market and Video Analytics

The industry of video surveillance was born in the 1950s when the first analog cameras were installed. Color cameras were introduced during the 1970s and ten years later technology which enabled video to be recorded was created. In 1996 Axis introduced the world's first network camera which started a slow technological shift from analog to digital surveillance (Axis Communications AB, 2011a). In their report the IMS Research Institute (2012) announced the total video surveillance market to be worth over $10 billion in 2011, and predicted it to double by 2016. Network video surveillance accounted for about 11% of the total market in 2011, but is forecasted to account for 60% of the total market in 2016 (Axis Communications AB, 2011a).

The security market is driven by increased prosperity, urbanization and a general trend towards greater security considerations (Axis Communications AB, 2011b). One contributing factor to market growth is the increasing security concerns triggered by for example security breaches, terrorist attacks and increased theft and robbery worldwide (Research and Market, 2012). Governments are passing regulations to install cameras at public places such as schools and hospitals and increase the possible adoption of video surveillance (OECD, 2004)

5.1.1 Video Analytics – a Common Type of Third Party Application

A majority of third party applications are so called video content analytics (video analytics) applications. Video analytics refers to the capability of automatically analyzing video content. Within the security industry this has historically been used for motion and objects detection, wrong-way detection and loitering detection. The application of video analytics has recently also been used within the retail industry where stores have started to use business intelligence applications to monitor queue lengths, count number of customers and map their way through the store. According to IMS Research Institute (2012), the interest for intelligent video applications is increasing and the potential of video analytics within non-security applications are dawning. Applications could for example be used within advertising where a camera can recognize the age and gender of a person or within the automotive industry where applications could detect driver drowsiness and assess the danger on the road.

5.2 Axis Communications AB

Axis was founded in 1984 as a company delivering print servers, but has over the years moved to become market leader within network video and network video surveillance cameras. Today Axis' profits are mainly related to sales of camera units. They provide network video solutions for professional installations featuring products and solutions that are based on innovative and open technology platforms. They are based in Lund but are acting globally with offices in 41 countries and partners in more than 179 countries. In 2012 Axis annual turnover was approximately 4 000 million SEK with 1 400 employees. (Axis Communications AB, 2013)

Axis was founded upon a distinct two-tier business model (see Figure 3), in other words an indirect sales model. By initiating partnerships Axis believe to have found a scalable solution with specialist expertise in each step and proximity to end customers. In order to deliver complete solutions to end customers Axis collaborates with several different actors such as distributors, system integrators and technology vendors. These are divided into four partner programs: channel partner, architecture & engineering (A&E) partner, technology partner and application development partner (ADP). With this master's thesis focusing on software only the ADP program will be described in more detail in section 5.3.2.

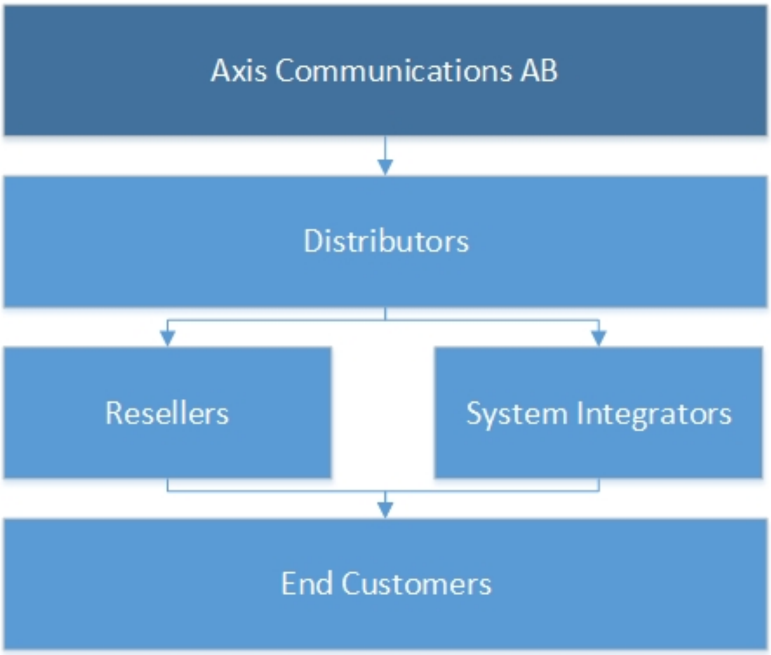

Figure 3 Axis' two-tier business model

5.3 Classification of the Software Ecosystem Surrounding ACAP

The following section consists of a presentation and classification of the software ecosystem surrounding ACAP in accordance with the theory presented in section 4.4: base technology, accessibility, extension market and network effects. The SECO is also visualized in Figure 4 below.

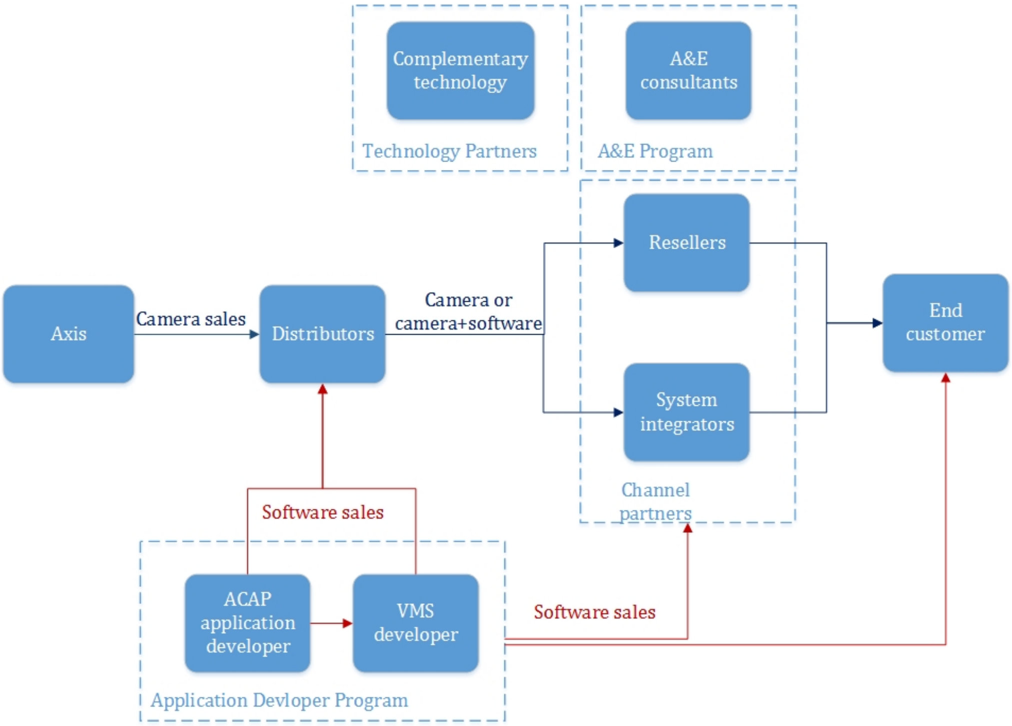

Figure 4 the software ecosystem surrounding ACAP

5.3.1 Base Technology of the ACAP SECO – an Application-Centric SECO

The software ecosystem in focus throughout this master's thesis is based on AXIS camera application platform (ACAP) which is 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. It was created to extend the functionality of the camera and meet specific end-user needs (Axis Communications AB, 2009). ACAP was originally only installed on some of Axis' products but are now available on the majority of Axis' different product lines. In order to enable and facilitate development towards the platform Axis also provides:

- An API with focus on communication with ACAP and external software.
- A Software Development Kit (SDK) for development of embedded applications. The SDK contains building environments and scripts, a

compiler system, API:s to access on-board device functionality, C-libraries, analytics libraries, C++ support , debugging and profiling tools.

- Compatibility tool to guarantee functionality of Axis' network cameras and video encoders. Compatible applications are listed on the Axis' web site.
- Optional copy protection tool to ensure that all applications in an installation are properly licensed. (Axis Communications AB, 2013b).

In section 4.4.1 Jansen and Cusumano (2012) described different types of platforms where ACAP is best categorized as a software platform in contrast to a software service platform or a standard. The two latter are platforms that are not de facto installed on a product whereas ACAP is installed on a physical camera. Using the classification taxonomy presented earlier (see section 4.4.1) presents ACAP as an application-centric SECO. The platform is founded on product lines that have already achieved success in the marketplace without a SECO, i.e. the platform offers customer value without third party applications in contrast to an operating system-centric SECO. Furthermore, the applications are domain specific, i.e. they extend the functionality offered by the platform. ACAP also shows some similarities with an operating system-centric software ecosystem in the offering of development tools and the sales of devices being highly influential. The similarities with an application-centric software ecosystem is however larger. Related research made the same assessment regarding a SECO of embedded software in the car industry (Eklund and Bosch, 2012) which supports the ACAP SECO being labeled an application-centric SECO.

5.3.1.1 Why Should Developers Join an Application-Centric Software Ecosystem

After the classification of the ACAP SECO the authors found a need to incorporate previous research on why developers join this type of SECO:s. Therefore theory on why developers join an application-centric SECO is presented in this section. This research provides a frame of reference in contrast to the findings of this study.

Earlier research by Bosch (2009) identified four success factors to make a platform a lucrative choice for third party developers:

(1) A large set of customers with a real reason to extend the platform functionality with third party applications. Previous research found that the platform leader's large set of customers is a primary factor for third party developers to develop towards a platform (Huang et al., 2009, Bosch, 2009, Zhu and Iansiti, 2012), since compatibility is often a prerequisite to access this aforementioned customer base (Ceccagnoli et al., 2012). An application-centric software ecosystem is built upon the success of a product before its transformation to a platform (Bosch, 2009). Third party developers need to know that the platform leader's customers actually have a need for the extended functionality of their applications (Bosch, 2009), since the customer's purchase not necessarily was made with third party applications in mind.

(2) Simplified development. Bosch (2009) identifies simplified development as an important factor for third party developers. They seek to maximize their profit (Zhu and Iansiti, 2012). Simplified development enhances their incentives by reducing the costs to attain the aforementioned pool of customers. Development is simplified by good developer environments and stable interfaces and is influencing factors for creating an attractive developer environment (Bosch, 2009).

(3) Seamless customer experience. The available resources to create a seamless user experience are an important factor for development (Bosch, 2009). This is also found in related research by Ceccagnoli et al (2012) who highlights the value of software interoperability. The difference from (2) is that these tools create a seamless integration in the eyes of the *customer* for example through the same user experience framework.

(4) Viable market channel. With the largest influencing factor on third party developer's incentives being the set of potential customers (see (1) above) exposure to them through a viable channel is of importance (Bosch, 2009). An example is to create an extension market as presented in section 4.4.3.

5.3.2 Accessibility of the ACAP SECO

In order to get access to the ACAP SECO a company has to be a member of Axis' application development partner (ADP) program. It was introduced in 2000 and enable third party software developers to extend functionality by integrating their solutions with Axis products (Axis Communications AB, 2013b). Being a member is free and requirements for a membership are low. The only screening factor is whether or not the applicant is a registered company or not.

There are currently three tiers in 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 companies have to prove that their solutions generate a certain amount of camera sales and that they have an active engagement with Axis. Taken from Axis' own description companies have to "... successfully integrated their commercial applications with a significant portion of the Axis product range...". To advance to the lately added Gold ADP level Axis requires "that Axis is their [the company's] preferred network video hardware vendor by meeting volume channel license requirements, sharing product roadmap information, dedicating development resources for integrating new Axis products and features, and regularly engaging with Axis business development managers". (Axis Communications AB, 2013a)

5.3.3 Extension market of the ACAP SECO

This software ecosystem has a list of extensions, in this case third party applications. This list is available on Axis' website and it is entirely controlled by Axis. Axis is however not handling any sales or transactions. Customers are referred to developers' websites in order to obtain more information or purchase applications. Axis is offering an optional licensing system which also could be seen as a part of the extension market. Both of these are freely available to all ADP members. Axis is not offering any joint way of purchasing third party application which forces third party developers to sell their software in other ways. This flow of sales is included in red in Figure 4 above.

5.3.4 Network effects in the ACAP SECO

Currently 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. Hence no direct network effects are observed. This is mainly due to the current performance of Axis' cameras which generally only allow one application to be run at once. This also lowers incentives for developers to collaborate because they cannot co-create value through utilizing each other's applications. Networking effects could hence be assessed as low.

5.4 Actors of the ACAP SECO and their Relationships

In this section relevant actors within this SECO are presented in more detail. Their main tasks are described and they are classified according to the model presented in section 4.3.1.

5.4.1 Platform Leader

Axis is classified as being the platform leader of the ACAP SECO. They are in control of the development of the underlying platform and their actions are highly influential on this ecosystem.

5.4.2 External Actors

Video Management System (VMS)-developers represent a large bulk of the ADP program. They develop external products, running on servers or similar, and most of them view image output and control cameras. Some of these video management systems utilize functionality of ACAP applications through incorporating them into their own program.

ACAP developers are third party developers who develop applications towards ACAP. On Axis' website there are currently 16 companies offering third party applications. Together they provide 27 applications (Axis Communications AB, 2013b). Many of these developers can be described as niche players i.e. they are smaller actors who target a specific market segment. ACAP developers are a part of Axis' ADP program. Most of these companies currently develop video analytics applications which are further explained above in section 5.1.1.

5.4.3 Vendors

System integrators (SI) and resellers are a part of Axis' channel partners program. They are companies who usually sell directly to end customers. The differentiating factor between resellers and system integrators is that the latter install and integrate Axis' products, while the former has no involvement in installations. These companies come in a wide array of sizes from small and local too large and global. This group is classified as vendors due to them making money on selling products produced by the software ecosystem. In case of large project the SI:s or resellers notify Axis through their customer relationship management program in order to receive price rates and information about distributors. Small projects are handled directly through distributors without Axis' involvement.

5.4.4 End Customers

End customers to Axis cameras are located all over the world with North and South America represent 50% of Axis' total sales, Europe, Middle-east and Africa 28% and Asia and Australia represent 12%. Furthermore camera customers are categorized into eight main categories; retail, transportation, healthcare, banking, city surveillance, critical infrastructure, education and industrial. Most commonly an end customer is a large institution such as a government, municipality, public transportation company, airport or a private operator. In many cases installations are large and complex. All camera customers are theoretically potential buyers of ACAP applications. There are no public figures of the ACAP application's distribution geographically but currently they are mainly within the same eight areas as the cameras. Resellers and SI:s sometimes work as customers, but not end customers, to ACAP developers.

5.4.5 Others

As Axis work with only a few selected distributors in each country and generally use large global companies. Distributors hold large camera volumes and cover credits to system integrators and resellers. The majority of distributors do not incorporate software into cameras before selling them and therefore they cannot be classified as vendors (Manikas and Hansen, 2013). Due to third party applications being so reliant on camera sales they are however highly affecting this SECO. A&E Consultants are companies that often are hired by an end customer to create requirement specifications that are applied to find suitable products and negotiate deals. These companies are often small and do generally not sell any solutions themselves.

6 Interview Results and Cause and Effect Analysis (RQ1 and RQ2)

This section presents the interview results connected to research question one and two. The reader is first presented with overall information about the interviewees and their companies in Table 1. The findings connected to research question one and two are presented in Table 2 and Table 3 in Appendix A with identified relevant findings further elaborated below. Finally, a cause and effect analysis is performed on the interview findings.

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

Table 1 Companies, interviewees and their roles

6.1 What are the Reasons and Benefits to Join and Participate in a Software Ecosystem? (RQ1)

Table 2 in Appendix A presents answers connected to why the company started and continued to develop towards ACAP. Mentioned benefits and advantages of ACAP are also included in order to provide more information about the perceived positive aspects of ACAP. Reasons stated by one single developers are also included in Table 2 but they are not given ID:s [1]and are not elaborated any further.

Two reasons were identified as having a substantiated effect on why third parties choose to join or participate in this software ecosystem: an end customer demand (given the ID Re1) and the relationship with Axis (given the ID Re2). Additionally two common benefits were identified: internal standardization (given the ID Be1) and external standardization (given the ID Be2). These are, together with connected deviations, further explained below followed by an analysis of root causes (see section 6.3).

6.1.1 Reason 1: End Customer Demand (Re1)

End customer demand (Re1) refers to an end customer demand communicated to a third party developer. This question (reason for developing) is not applicable when it comes to Company C and E because they have not developed any applications to ACAP. Three out of four (Comp. A, B, D, not F) of the interviewed companies that currently do develop towards ACAP express that one reason they had an initial interest in developing applications on Axis' cameras, was that they received information about an end customer demand.

In the case of Company A they received a clear demand from a customer that their cameras had to be integrated with Axis' cameras to be a viable alternative and this was according to A1 the main reasons for them to start development. 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 D1, head of the video analytics development group at Company D, they received information of an end customer demand by their sales department.

The fourth company currently developing towards ACAP (Comp. F) did not mention the end customers' needs as a reason for them being interested in ACAP. One reason for this could be that they, unlike the others (Comp. A, B and D), are not developing video analytics applications but video management systems. They stated that it was very easy for them to convert their solution, indicating a lower perceived risk, which might have reduced the need for a clearly communicated end customer need.

[1] The most common findings were given ID:s, presented in Appendix A, table 2-4.

Even though they (Comp. F and D) did not mention it as a reason they mentioned customers benefitting from less need of resources or infrastructure with embedded solutions. According to F1 embedded solutions, as ACAP applications, could reduce the amount of resources, space and money needed by the end customer due to the absence of servers. Servers take up a lot of space and are both expensive to purchase and maintain. These smaller and cheaper solutions are according to F1 suitable for smaller more remote places which enable them to create additional value both to new and existing customers through offering a more resource efficient solution.

6.1.2 Reason 2: Relationship and communication (Re2):

The relationship and communication with Axis were discussed during all interviews. In some interviews (Comp. A, B, F) these were mentioned as a reason for developing but in most cases it was discussed in a more loose form. This subject was identified as an important factor in why companies join or participate in this software ecosystem. Extra attention is therefore given in this upcoming section. All discussions and statements were compiled and are presented in Table 3 in Appendix A.

All companies in this study that have an ACAP-product available (Comp. A, B, D, F) did have a relationship with Axis prior to starting developing towards ACAP. All four describe this relationship as good with three out of four (Comp. A, B, F, not C) emphasizing transparency and openness. Among the companies not developing towards ACAP (Comp. C, E) this was not expressed, in one case (Comp. E) the total opposite was described, they perceived Axis as being opaque. They experienced difficulties with finding the right people to talk to and to get clear answers about technical specifications, future development and customers.

Connecting back to Re1: In order to identify a customer need a company has to have or collect knowledge about its potential customers. Company A, D and F were already in contact with potential customers within the security industry as they had the same or a similar application available on another platform before applying it to ACAP. They already had an established relationship with Axis. Company B on the other hand did not have any information about Axis customers initially. They received informal information from Axis and through having a personal relationship they received information they would not have received otherwise. This implies that a personal relationship (Re2) with Axis, who has knowledge about their own end customers, facilitates receiving information about direct end customer needs. This is also supported by the fact that Company E who states that they have a hard time getting answers from Axis and describe them as being opaque chose not to develop towards ACAP. They also mention lack of information about customers (He2) as a hesitation towards developing, which supports the importance of Re1 as reason for developing towards ACAP.

6.1.3 Benefit 1 and 2: Internal Standardization (Be1) and External Standardization (Be2)

Two different types of standardization are referred to during the interviews. First, standardization within Axis' product line which enables the same code to be used in the majority of Axis's cameras. This is onwards referred to as "internal standardization" (Be1). Secondly, external standardization in communication between the camera and external software will be referred to as "external standardization" (Be2). An external standard does not exist today, but the perceived possibility of Axis implementing it is seen as a benefit of ACAP.

6.1.3.1 Internal Standardization (Be1)

Three out of four companies developing towards ACAP (Comp. A, B, D) mention internal standardization as a benefit (Be1). This is supporting previous research on the importance of simplifying development for third party developers (see section 5.3.1.1). The fourth company (Comp. F) mentioned it as a reason. A reason for Company A, B and D mentioning it as a benefit and not a reason could be that they already offer applications to Axis' cameras prior to ACAP which Company F did not. Prior to ACAP all embedded solutions had to be individually modified to each camera product line. This demanded both a lot of time and resources. When Axis launched ACAP this was well received by Company A. They experienced a decrease in efforts put into modifying their code. The same benefit was mentioned by interviewee B1. The companies not offering any ACAP applications (Comp. C, E) did not mention internal standard as a benefit of ACAP. This might be because they have not gone through the process of developing applications prior to ACAP and do therefore not experience the perceived benefit.

6.1.3.2 External Standardization (Be2)

Two of the companies in this study (Comp. D, E) mention external standardization (Be2) as a benefit. D1 expressed it as a common way to communicate to the outside world. D1 also states that this would enable Company D to sell their analytics solution separate from their VMS. Both D1 and D2 said that 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. E1 also see great benefits in an external standard. They explain that by having a standardized way of communicating with external software they would not have to work individually towards each VMS, but could instead focus their work on the camera and in others words use fewer resources. Additionally D2 describe that in the future they hope that ACAP will decrease the need of infrastructure for end customers and hence reduce the need of local support due to easier installation and increased compatibility, they explain that this would widen their network and distribution of their software.

Company A does not mention external standardization (Be2). One reason for this could be the fact that they use ACAP as an intermediary solution where their application runs partly on ACAP and partly on a server with the latter facing other systems i.e. an external standard would not be beneficial to that solution. Company B offers business intelligence solutions and Company F provide their own VMS, hence their ACAP application (Comp. A, B, F) do not communicate with external software. They would therefore not benefit from a standard for communicating with external software.

6.2 Why do developers hesitate toward joining or participating in a software ecosystem? (RQ2)

Hesitations against starting to develop ACAP applications are presented in Table 4 in Appendix A. Mentioned drawbacks of ACAP are also included in order to provide a complete picture of the perceived negative aspects of ACAP. Hesitations stated by one single developer are also included in Table 4 but they are not given ID:s and are elaborated any further

6.2.1 Hesitation 1: Axis' Business Model (He1)

Two companies (Comp. C, E) mention Axis business model, see section 6.2.1 , as a main reason for them not to develop towards ACAP. Company E explains that their largest issue is always to find new customers and that requires information about the end customer, information they find it difficult to acquire from Axis. Company E further elaborated on that Axis two-tier business model is focused on large distributors who would not be interested in working with company E. Company E prefers to work directly with system integrators or VMS companies through an OEM business model, where the VMS company includes company E:s software into their solution. Company C also prefers an OEM business model. Company C explains that it is not their perception 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. These two companies (Comp. C, E) are not developing towards ACAP and they are also the companies that do not describe having a good relationship with Axis (Re2). This could indicate that good communication (Re2) with Axis facilitates working with Axis' business model.

Company A and B already developed embedded applications to Axis cameras before ACAP which means that they most probably already had aligned their business model with Axis'. Another explanation of them (Comp. A, B, D, F) not mentioning Axis' business model as a drawback could be because they already had an existing product within that industry before ACAP. Hence they already had established sales channels which might have been applicable to ACAP products as well.

6.2.2 Drawback 1: Performance (Dr1)

Company A and B did not experience any hesitation when adapting their applications to ACAP. They stated that almost everything was an improvement. Because they were already involved in development of embedded solutions in Axis' cameras, ACAP only provided benefits and made their job easier. However, A1 do mention the performance of ACAP and more specifically the processor as being a drawback (Dr1). Axis does currently not allocate any protected amount of processor power to third party applications. A1 describes that in some situations this results in Axis' own internal processes grabbing too much power. However, companies will always welcome a more high performing technological solution providing them with for example more processing power. As B2 said: "Regardless of how good the processor becomes we would be able to improve the performance of our application and use the entire capacity […]".

Currently Company C is not able to adapt their product to ACAP because the functional features they need are not efficient enough. They mention several technical capabilities they wish Axis to adapt in the future to better fit their solution. D1 did not initially believe that the platform would have enough capacity to be able to run their applications, but that was not the case. However there is some difference in performance between cameras i.e. all cameras are not performing equally. This is a drawback that causes annoyance according to D1. D2 further describes that they are making their code more slim and efficient in order to implement their server solutions on ACAP. The need to make the code more slim and efficient for implementation could be a factor for all companies, as all applications have been available on servers first and then ported to ACAP.

6.3 Cause and Effect Analysis

In order to find out which contextual factors of the ACAP SECO that affects participation a cause and effect analysis was conducted. The effect, i.e. the reason for development, was put on the right hand side of the fishbone diagram (see Figure 5, Figure 6 and Figure 7). Seven contextual factors and their subgroups (described in section 4.3.1 and 4.4) were used as starting points to find potential causes. Theory was then combined with collected empirical data (both from external and internal interviews) in order to identify specific underlying causes that affected the identified reasons. The horizontal lines are identified causes which in some cases are broken down into sub causes. The causality is described under each diagram.

6.3.1 End Customer Demand (Re1)

Development based on an end customer demand is commonly called a market pull strategy (Brem and Voigt, 2009). The contrasting strategy is called technology push, meaning the software is first created then a customer with a fitting need is found or created through marketing (Brem and Voigt, 2009). The cause and effect analysis was not suited for this identified reason. It did not provide an understandable view of the underlying reasons. All developing companies in this study had previous applications that they ported to ACAP. As they already had a solution available the customer had an ability to request its implementation on the new platform. By already being in the industry the companies already have a network of customers, decreasing the need to access Axis existing customer base.

6.3.2 Relationship and Communication (Re2)

There are many contextual factors within the ACAP SECO that increases the need for good relationship and communication with the platform leader. The most prominent causes are found to be need of trust, accessibility to customers and information. The causes are presented below in Figure 5.

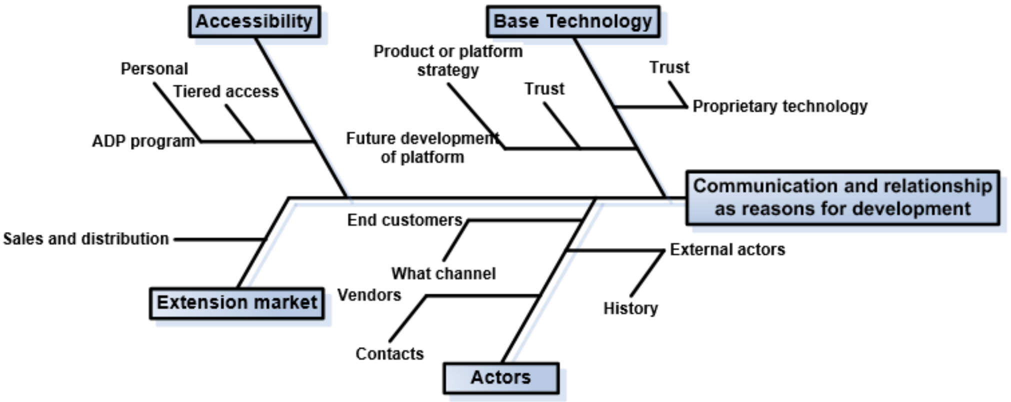

Figure 5 Cause and effect analysis for communication and relationships as a reason for development

Base technology: Developing towards ACAP means making customizations towards Axis proprietary technology. Making investments in technology that cannot easily be redeployed is commonly called an asset specific investment (Williamson, 1991). Axis is are in control of the proprietary technology of the platform and are hence in total control if its development. Therefore developers have to trusts Axis to develop the platform in an acceptable direction. The need of trust is identified as a possible cause for communication and relationship being an important part of the SECO as it can be seen as risk reducing (Teng and Das, 2001). ACAP is a very small part of Axis current business. Developers without trust in Axis would probably be wary of Axis focusing on the basic camera features rather than developing the platform, as indicated by Hagel et al (2008). The difference between the product- and platform strategy could cause uncertainties which also could be reduced by an increased amount of trust built from relationships and good communication.

Accessibility: Axis ADP program offers tiered access to business and technical information. But as presented during the internal interviews at Axis the highest tier is not commonly reached by ACAP developers. But through personal relationships at Axis the information of the higher levels can still be attained causing communication and relationship to become a reason for developing (Re2).

Extension market: Providing a viable market channel is a large influencing factor on developer's incentives (see section 5.3.1.1). As presented in section 4.7.3.3 Axis does not provide a way of selling or distributing ACAP applications. Therefore developers without an established customer base or a relationship with Axis would experience a more uncertain way to market, which is further elaborated in section 6.3.4.

Actors: The developer's history with Axis (the platform leader) was after the interviews with external developers found to be a cause. It is the authors belie that 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.

6.3.3 Internal Standardization (Be1)

Internal standardization was described as a benefit or reason for development towards ACAP. In this SECO there are multiple factors that cause this to be a benefit or a reason, presented in Figure 6. Internal standardization means that the same code can be reused on different types of Axis' cameras. The effects are similar to that of software product lines and the reuse of code (Eklund and Bosch, 2012), making development to several configurations more efficient in terms of cost and risk (Niwot Ridge Consulting, 2001).

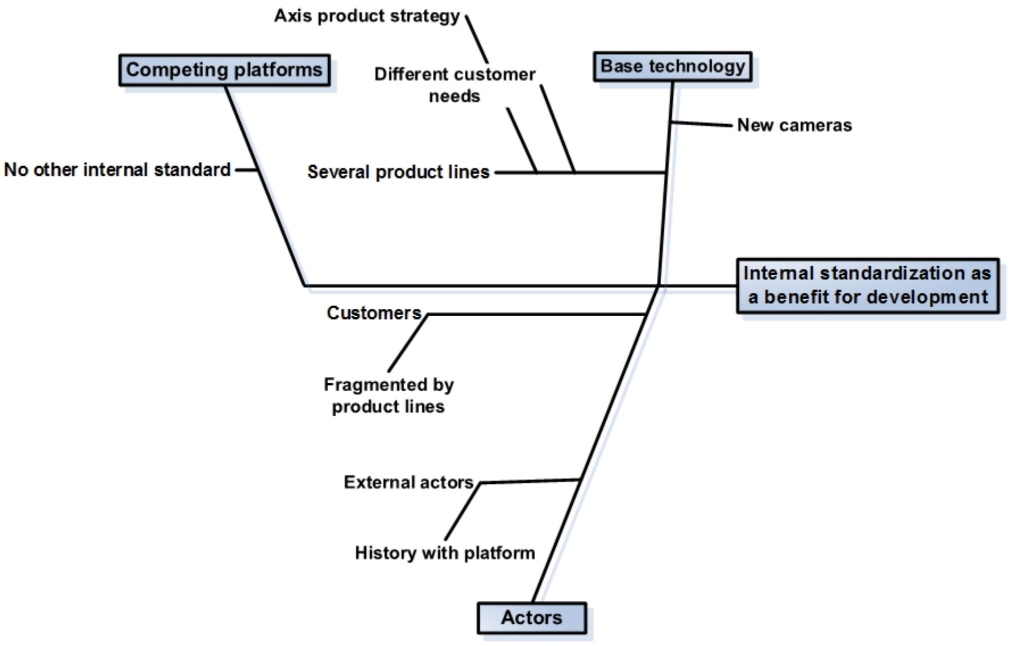

Figure 6 Causes for internal standardization as a reason for development

Base technology: Axis offers several camera product lines with different hardware configurations and ACAP is installed on these product lines. Axis provides several product lines because they fulfill different customer needs. This is an important part of their product strategy. An internal standard becomes important to third party developers as fewer configurations are needed to work with all type of cameras.

New cameras have historically not been created with the purpose of improving the utilization of third party applications, but improvement of the cameras basic features has rather been the goal (revealed during internal interviews at Axis). As new cameras are released, there is a risk that third party applications will not work without an internal standard.

Actors: The customers, either the end customer or vendor, 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 history of the third party developer was deemed an important cause because developers, who developed software prior to ACAP, without the internal standard, saw a greater benefit in it.

Competing Platforms: For internal standardization the fact that no competing platforms are offering an internal standard for embedded software in their cameras was identified as an important cause. This causes developers in need of it to be more interested in ACAP.

6.3.4 External Standardization (Be2)

External standardization is the ability to more easily integrate ACAP applications with other software, through for example standard API:s. As an external standardization is not fully implemented, the developers see a future potential of its implementation. Multiple contextual factors of the ACAP SECO were identified as causes for external standardization as a reason for development (see Figure 7).

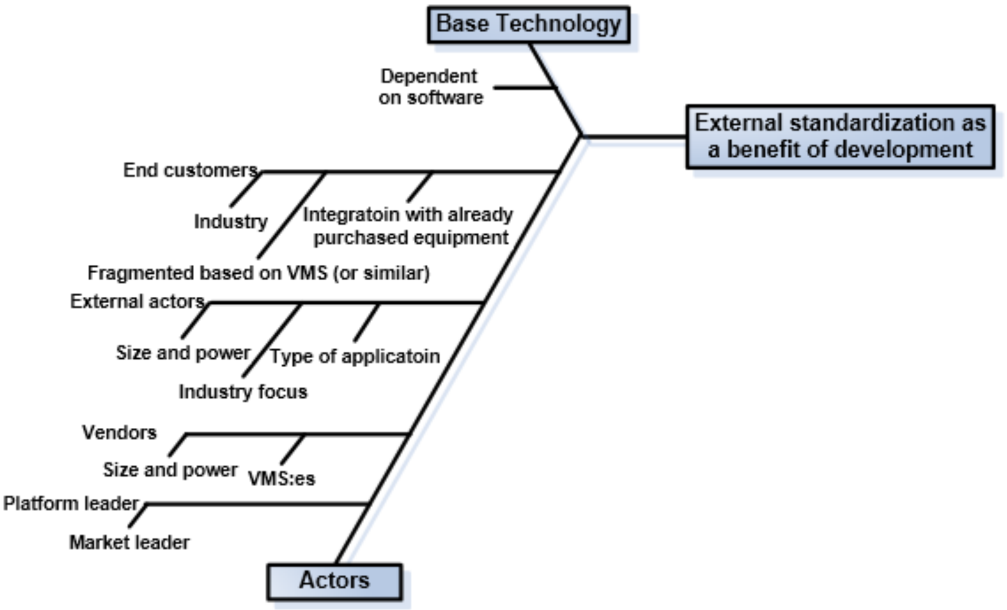

Figure 7 Causes for external standardization as a reason for development

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 the application in order to use it. This interpretation requires customization between the ACAP application and the external software. Therefore ACAP developer become dependent on external software being able to interpret the data they send. An external standard would reduce that dependence and the need of such customization.

Actors: A large amount of causes were found among the actors this SECO. Many of these causes indicate that because ACAP applications are niched they need to be compatible with other software to provide a complete customer solution (as described above in base technology). Due to the amount of different types of such software, each with their own API:s, the barriers of entry are quite large without an external standard.

Depending on the industry the customers are fragmented by their use of different VMS:s or similar controlling software. In order to reach customers who already have an installed solution ACAP developers have to be compatible with that equipment. A higher compatibility with those systems may decrease the barrier of entry and increase the potential customer base.

External actors, i.e. the ACAP developers themselves and for an example VMS:s are also a reason behind the importance of external standard. ACAP developers are commonly niched video analytics companies and are often small compared to VMS developers. Due to their size their dependence towards others might increase. Hence ACAP companies have to customize their solution to fit every VMS whose customers they want to reach. Some types of ACAP companies do however not need the external standardization. Such companies provide their own VMS or their application is integrated through other means.

Axis is both the platform leader and the market leader within network video solutions. That makes them an influential actor in this SECO. An external standard is not fully implemented but still provides a benefit of ACAP. The cause for this seems to be that Axis is the market leader and thus is the only one that has the potential to implement it.

6.4 Synthesis of RQ1 and RQ2

Developers join Axis software ecosystem because of an end customer's demand. ACAP enables the customer to better utilize already purchased equipment (i.e. an ACAP enabled Axis camera) or purchase less infrastructure in a new installment. This creates higher customer benefit through opening new markets by offering reduced cost, less infrastructure and higher efficiency.

An internal standard becomes important due to multiple product lines being offered and adopted by the end customer. Through an internal standard the developer can reach a larger customer base at a lower cost. This is not offered by competing platforms and is especially valued by developers who developed applications to Axis cameras before ACAP was introduced. An external standard would decrease the developers need to individually collaborate with multiple partners to ensure compatibility. Through an external standard more customers can be reached with fewer resources. ACAP applications need to be compatible with other software, both to function and to reach the market, was very prominent. An external standard is however not fully implemented but Axis, as the market leader, is seen as having the potential to implement it. The type of application and the type of industry the company focus on was identified as influential on the company's perceived benefits and their reasons for participation.

The third party developer's relationship with Axis, and how they perceive their communication, was identified as a reason to why companies developed towards ACAP. The need of trust and a personal relationship to overcome market uncertainties were factors identified to be a possible cause for this reason. Through a better relationship with Axis information about possible markets and distribution channels can become attainable. Axis' business model was however described as a hesitation by the two companies that have chosen not to develop. Both of these companies had issues with responsibility of support and training if they were to sell their software directly to end customers. One of the non-developing companies also found it difficult to assess the business opportunity of the SECO as they did not know where the potential customers were to be found. Finally performance was perceived as a drawback and a hesitation.

7 Governance Activities Performed by Axis (RQ3)

Information about Axis' activities was retrieved during the initial exploratory interviews, informal meetings with relevant employees, an internal survey and from documents at Axis intranet. The activities collected from theory (see section 4.7) were compared to Axis' current activities. Each activity is marked as [YES], [NO] or [PARTIALLY] depending on to what degree the activity is performed.

7.1 Activities Connected to Niche Creation

7.1.1 Expand applicability [YES]

A purpose of ACAP is to expand the applicability of Axis' cameras to increase sales. Axis are expanding the applicability of the platform by providing access to new features and by releasing more powerful cameras created for new environments. Therefore this activity is assessed as being performed.

7.1.2 Make strategy explicit [NO]

None of the interviewees received information about Axis' strategy of ACAP explicitly. Those answering yes to being aware of Axis' strategy (Comp, A, B, D) stated that they received it during discussions and collaboration with employees at Axis i.e. implicitly. Axis does not have an explicit strategy for ACAP.

7.1.3 Create API [YES]

Axis has created a collection of API:s connected to ACAP.

7.1.4 Co-development [NO]

Axis does not perform any co-development i.e. joint development projects with other companies and therefore this activity is assessed as not being performed.

7.1.5 Develops new business models [NO]

Axis' main focus is on camera sales. In the case of the cameras they never deviate from their two-tier sales model but they do not have any similar requirement for software sales. Axis provides a free licensing system to the users of the platform. However, Axis is not in other ways involved in the sales and distribution of ACAP applications. Axis is restricting the use of certain business models when not allowing third party developers to be a part of their chain of distribution. An example of such business models are the ones based on OEM. They do not perform co-development which also would have been a way of create new business opportunities and business models for third party developers and they do not offer a central marketplace.

7.2 Activities Connected to Robustness

7.2.1 Create partnership model [YES]

Axis' ADP program is an established partnership program offered to all companies interested in developing software connected to Axis' cameras. The program consists of three levels although the only screening criterion for the entering level is to be a registered company. It is not exclusively designed for ACAP developers, but rather towards all software companies wanting access to connect or adapt their software to Axis' cameras. Partners receive benefits not available outside the partnership such as pre-release product information, co-marketing opportunities, pre-launch support and developer training. Axis is therefore assessed as performing this activity.

7.2.2 Do marketing [YES]

Axis main marketing activities are conducted in order to increase the sales of cameras. ACAP is however marketed as a benefit of the camera. The features of ACAP are promoted at fairs and exhibitions, mainly within the security domain. ACAP applications are exposed as camera applications on Axis website. Marketing activities towards potential ACAP developers are sporadic and small compared to the marketing of cameras.

7.2.3 Partner development programs [YES]

Axis learning center provide training, seminars, classroom training, tools and quick reference help (Axis Communications AB, 2013c) and is accessible for members in the ADP program. It is not designed as a program but rather as a source of information, support and training. Axis does not offer any financial support. But the main reason for a development program is to help strengthen members of the ecosystem and that is fulfilled today.

7.2.4 Form alliances [PARTIALLY]

Axis has existing alliances with many relevant companies within the industry through their partner programs (ADP partners, channel partners, A&E partners and technology partners) but the focus of these relationships are not on ACAP or its applications. But because they have these existing alliances which could be utilized in the work towards strengthening ACAP and its SECO this activity is assessed as partially performed.

7.2.5 Stabilize API:s [YES]

Axis have stable API:s. New features have been added since ACAP's introduction, but backwards compatibility is ensured as no API:s has been changed. Axis is aware that the API:s are not optimal, but sees it as a higher priority to keep them stable rather than changing them. Findings from the survey indicate that the API:s are more stable than the developers expected.

7.2.6 Raise entry barriers [NO]

Axis does not impose high entry barriers to joining their ADP program: members only have to be a registered company. Furthermore, they do not take any fee or commission associated with published applications. The compatibility of applications is tested but not their performance hence they are not taking any direct measures towards raising the entry barriers of their ADP program-Nevertheless, they do not allow independent developers access to the ADP program, for example students, acting as an entry barrier for that type of developer.

The cause and effect analysis in section 6.3 revealed several factors that raised the overall entry barriers of the SECO: the dependence of external software and other actors, the fragmented customer base of Axis' end customers and the lack of an accessible way to reach them causing an uncertain way to market. Additionally the low number of participants implies high entry barriers. The entry barriers are not assessed as being managed by Axis directly. Therefore this activity is still marked as not performed.

7.2.7 Propagate operation knowledge [NO]

Axis does not have a systematic way to collect end user experiences or feedback related to ACAP and is hence not able to communicate it to other members of the SECO. Therefore this activity was assessed as not performed.

7.3 Activities Supporting Productivity

7.3.1 Organize developer days [NO]

Axis has organized developer days in different forms and with different focus throughout the years. Before launching ACAP they held a training session in Lund. However, the current arrangements do not include ACAP developers, unless they offer an additional product as well and hence are qualified.

7.3.2 Collaborative marketing [PARTIALLY]

Axis does not systematically perform collaborative marketing efforts with third party developers. On a case by case basis some forms of collaborative marketing is however performed at exhibitions and fair. As it is unsystematic and seldom performed, this activity is assessed as partially.

7.3.3 Create sales partner program and create new sales channels [NO]

Axis has a channel partner program including companies distributing and selling network video products and solutions (Axis Communications AB, 2011a). This does not apply to distribution of software or more specifically ACAP applications. Axis does not have an outspoken strategy for how ACAP applications should go to market. This activity is hence assessed as not being performed. However, Axis have historically seized opportunities to cooperate with existing customers and then provided information and sales support. Though, this is done sporadically and through personal connections.

7.4 Synthesis RQ3

Axis is currently expanding the applicability of their platform but they are not communicating their strategy or vision of ACAP explicitly. By creating API:s Axis opened up their platform. The stability of the API:s is highly valued in order to keep existing applications compatible. Axis is providing third party developers with support in marketing, sales and development based on personal contact and relationships. Axis is however not performing co-development or facilitating new business models.

Axis has an established partnership program with three tiers. The program is not design for ACAP developers, but rather for larger software companies. Currently Axis has one requirement for companies joining their ADP program: that they have to be a registered company. Axis also offers an extensive learning program in order to strengthen the members of the SECO. They do not have an organized way of communicating operational knowledge or raise entry barriers within the SECO.

Current developing days, or similar activities, have not been focused on ACAP. Axis is infrequently including ACAP applications on exhibitions and fairs in order to market them, as well as the platform. Axis' sales partner program does not include ACAP developers. Information and sales support is rather based on personal contact.

8 Analysis

This section combines the findings of research question one, two and three with previous research. This is done in order to answer the purpose of this thesis.

8.1 Expand applicability

Theoretically, expansion of applicability should increase the variety of participants by making the platform a viable option for new applications and solutions within different areas. Because majorities of participants are active within the same industry and they are providing similar types of applications the variation of this SECO is assessed as low. This could indicate that the applicability of the platform is limiting developers in their use of the platform. This is supported by Company C not being able to create an application due to the platform not providing access to the features they need.

8.2 Make strategy explicit

There is no explicit strategy of ACAP which according to theory should create decreased trust among developers towards the platform leader. However, there were no trust issues identified among the developers participating in this study. One common factor of the companies choosing to develop were that they all had a previous relationship with Axis described as good and transparent indicating increased trust. Furthermore, several companies received what they perceived to be information about Axis' strategy implicitly through contact with Axis personnel. Hence, indicating that the need for an explicitly communicated strategy decreases for those with a relationship and good communication with Axis.

8.3 Create API

Axis has created API:s for ACAP which were found to be decreasing compatibility issues in the software ecosystem, supporting previous research by Jansen and Cusumano (2012). The API was described as a benefit and a reason to develop towards the platform (Be1: Internal standardization). During the cause and effect analysis in section 6.3.3 three factors were identified as causing an increased need for an internal standard in this SECO: (1) base technology: several product lines, (2) actors: fragmented customers and (3) competitors: not offering an internal standard similar to ACAP. These factors hence increased the perceived benefits of an API.

8.4 Co-development

The lack of co-development has not had any identified effects on this SECO.

8.5 Develops new business models

Axis does not have any requirement on how software is sold or distributed in the SECO. Theoretically any business model could be applied. However, only type of business model is used among the participants in this study. They are all using a license based business model. Axis offers a licensing system for free connected to the platform. However, not everyone was using Axis' licensing system but some had created their own. Axis is not facilitating any other types of business models. This indicates one out of two things. Licensing based business models are either common because Axis promotes it use or because it is a good fit for the environment of this SECO.

8.6 Create partnership model

The ADP-program is explicitly focused to promote developers of high volume and broad applications rather than niche applications, which most ACAP applications are. The requirements (presented in section 5.3.2) that have to be attained in order to reach the highest level makes it very difficult for ACAP developers to advance due to their size and niched applications. The ADP program is not identified as affecting the incentives of the participants of this study. However, it seems to create a paradox where ACAP developers need support to expand their business but in order to obtain this support they have to advance to a higher level, which they are excluded from due to lack of sales. Historically, corresponding benefits were accessed through a personal relationship causing a relationship and good communication to be more important for development (see section 6.3.2). Axis partner program does not allow independent developers, decreasing the variety of the SECO.

8.7 Do marketing

The developers in this study have started development due to end customers demanding the developer's product on the ACAP platform (Re1). This indicates that the customer has become aware of platform and its merits. If this awareness is due to marketing conducted by Axis, by another actor, or a combination is however unknown. The finding is however indicating that awareness of the platform among potential customers is of importance for developers in this SECO. The majority of ACAP developers already had a relationship with Axis before developing to ACAP. This suggests that the developer's awareness of the platform is limited to Axis existing network, limiting participation.

8.8 Partner development programs

Axis learning center influences the participation in the SECO by making the technical transition to the platform easier. The technical expertise delivered by Axis was found to affect the perceived quality of communication and relationship by the developers, which was a reason to join this software ecosystem (see section 6.1.2).

8.9 Form alliances

Axis alliances are not focused on ACAP. But their alliances within the security industry affect participation by creating a perceived possibility of Axis implementing an external standard (see section 6.1.3.2).

8.10 Stabilize API:s

As mentioned above in section 8.3 API:s were found to be decreasing compatibility issues in the software ecosystem. They were also described as a benefit and a reason to join and develop towards ACAP (see section 6.1.3.1). Stable API:s provide the same benefit over time; compatibility with new products as well as old. The perceived benefits of having an internal standard (Be1) are over time affected by the stability of the API:s. Underlying causes which increased the need for API:s are presented in section 6.3.3. There are indications that the API:s is more stable than the developers expected.

8.11 Raise entry barriers

According to previous research high entry barriers might exclude too many developers (van den Berk et al., 2010). Even though Axis does not directly raise the entry barriers of their ADP program, external factors seem to create perceived high barriers, see section 7.2.

A majority of ACAP developers are active within the surveillance industry. Additionally all ACAP developers in this study were active developers before starting to develop towards ACAP. This indicates that entry barriers are perceived lower by this type of actors. Consequently the probability of participants entering the SECO from outside this group are assessed as lower causing a domain dependence and lowering the variety of developers. The domain dependence together with the relatively low number of third party developers (low relative the number of Axis other partners and end customers) imply that Axis should facilitate participation and lower entry barriers in opposite to what is suggested by Jansen et al (2012).

8.12 Propagate operation knowledge

No effects of Axis not propagating operational knowledge were found. However, Axis does not collect operational knowledge from its end customers. Their two-tier business model reduces Axis' direct contact with end customers and affects their ability to collect such data. Therefore this task might not be suitable for the platform leader in this SECO.

8.13 Organize developer days

Previous research (see 4.7.3.1) identifies developer days as an opportunity to create awareness of the platform among potential developers and create bonds among participants. Axis is not organizing developer days, or any other activities sufficiently fulfilling this purpose.

In this SECO there is a low degree of interactions between ACAP developers leading to decreased connections, knowledge sharing and collaboration between the same. Axis does not facilitate developers meeting or interacting. The incentive of collaboration is further decreased by the performance of Axis' cameras only allowing one ACAP application running at once.

The lack of an external standard and the need of compatibility with external software (see section 6.3.4) increase the need for connections with other developers. These connections can be facilitated by developer days. The lack of developer days have however not been identified as directly effecting participation in this SECO. An explaining factor could be that current developers in this study were already offering similar solutions prior to ACAP and thus already had established connections to necessary external actors. A lack of organized opportunities to create connections would therefore mainly affect new participants without previously established connections.

8.14 Collaborative marketing

Due to Axis' unsystematic performance of collaborative marketing efforts no effects on participation were identified.

8.15 Create sales partner program and new sales channels

Due to Axis not having a structured sales partner program the possible increase of sales margins of ACAP software could not be evaluated.

The route to market for ACAP developers requires multiple relationships with other external actors and vendors for distribution and sales. As stated in section 7.3.3 Axis is not involved in sales or distribution of ACAP applications. Many ACAP developers are relatively small players in the surveillance industry which most often make them less influential compared to other actors as discussed in section 6.3.4. Historically, this has been overcome by ACAP developers receiving informal support based on a personal relationship. New developers without industry experience or a relationship with Axis would find it difficult to assess which relationships are needed to access the end customers. Hence, the lack of a sales partner program creates an inaccessible and uncertain way to market for new ACAP developers. A close relationship with Axis indicates a reduction of the perceived risk (Teng and Das, 2001) and facilitates access to important information and support. Hence, relationship with Axis becomes increasingly important to develop towards the platform (Re2).

9 Interpretation and Discussion

In this section the main findings from the analysis are interpreted and discussed. The reader is presented with an elaborated picture and the possible consequences of the findings.

The findings of this master's thesis indicate that both performed and not performed governance activities by the platform leader affect the participation in a SECO, and that the contextual factors set the stage for which activities that are most influential.

It is the authors' belief that many of the contextual factors found at Axis are common in other hardware focused industries. An example is the indirect sales model which has become very common in technology sales (Accenture, 2009). Thus the effects of the contextual factors found in Axis' SECO are likley to be found in those industries as well. The same is true for the conflicting strategies that were found to be common for application centric software ecosystems by Bosch (2009). It is the authors belief that this conflict would result in comparable unsystematic governance activities in other cases of the same type, thus affecting participation in similar ways.

As more hardware focused companies are showing an interest in developing SECO:s around their products it is likely that third party developers will gain more bargaining power. By understanding how the platform leader's governance activities together with the contextual factors affects participation companies can spend their resources better and make better decisions about their ability to create a SECO.

9.1 The Relationship

The third party developers' relationship with Axis, and how they perceive their communication, was identified as a reason to why companies developed towards ACAP. The most influential factors were found to be the lack of an explicit strategy for ACAP, the lack of a sales partner program and a fragmented and inaccessible customer base, which are further elaborated below:

In the cause and effect analysis (section 6.3) the need of trust and hence a personal relationship to overcome market uncertainties were factors identified to be a possible cause for a relationship and good communication being a reason to develop. Due to the relative size of ACAP developers compared to Axis, and that the development of an ACAP application is an asset specific investment, they need to trust Axis to develop the platform in an acceptable direction. Axis does not have an explicit strategy for ACAP which according to theory should decrease trust among developers. However the need for an explicit strategy seems to be reduced in this case where a personal relationship and a good communication function as a way to access this information implicitly.

The design of Axis' partnership model is not adapted for ACAP developers which make it difficult for them to advance to the highest levels and reach increased support and collaboration. Historically corresponding benefits have been accessed through a personal relationship causing a relationship and good communication to be more important for development.

Potential end customers, i.e. customers who own, use or purchase an Axis camera, are scattered all over the world and use multiple purchasing channels. The lack of a sales partner program or other ways to easily access end customers complicates ACAP developers way to market and make participation more difficult. It requires developers to have multiple relationships with other external actors and vendors for distribution and sales. Bosch (2009) stated that providing a viable market channel was seen as very important in order to increase participation (see section 5.3.1.1). Historically, this has been overcome by receiving informal support based on a personal relationship. Thus, the lack of a sales partner program and systematic development of sales channels for ACAP developers affects participation by increasing the need for a relationship with Axis in this software ecosystem.

Axis' unsystematic governance activities or the lack of governance, together with the characteristics of this SECO, create an environment where a personal relationship and good communication provide the developers with prominent benefits, and hence influence the choice of participation. When rules, procedures and a strategy for the future have not fully emerged relationships form an effective basis for trust. But if a relationship with Axis is a requirement for developers to perceive the benefits of this SECO sufficiently high to join, how should Axis act in order to increase participation? Is it a scalable solution and can it provide the value Axis is looking for?

Currently Axis is providing support and facilitates growth to members who contribute to a sufficient amount of camera sales (see section 5.3.2) i.e. companies who provide them with a high amount of direct value. The governance activities evaluated in this study are based on SECO health. A healthy SECO is described as creating sustainable opportunities to its members and create value for the entire SECO. Individual members may come and go while the ecosystem in large is preserved. A platform leader aiming for such a SECO is adapting a keystone strategy (Iansiti and Levien, 2004). A keystone provides its SECO with value and increase productivity, robustness and niche creation capabilities (Iansiti and Levien 2002). Common benefits created by this approach are co-evolution, increased innovation and collaboration (Bosch, 2009, Barbosa and Alves, 2011). In contrast to a keystone strategy a platform leader can adapt a dominator strategy (Iansiti and Levien, 2004). A value dominator often has little direct control of its ecosystem. A value dominator strategy may create extraordinary returns short term but is not seen as a long term solution (Iansiti and Levien, 2004). A dominator often contributes with relatively little value and will limit the SECO's possibility to grow. There are some similarities between Axis' performance in the ACAP SECO and the value dominator theory.

Axis want to expand their SECO but the area of software ecosystem in combination with a hardware product focus is new and quite unexplored. There is therefore a risk of them taking on a too dominating role in their efforts to utilize the SECO.

9.2 Internal Standard

Axis providing an internal standard was perceived as a major benefit by the developers interviewed in this study. During the cause and effect analysis the underlying reasons for this were that Axis provides several product lines and that end customers were fragmented by the use of these different product lines. Through creating API:s and keeping them stable Axis enable ACAP developers to reach a larger customers base at a lower cost and therefore offered a more attractive environment to participate in. This supports the findings by Bosch (2009) that simplification of development is an important factor. There were however a potential risk of Axis focusing too much on keeping the API:s stable instead of optimize their performance towards developers and thus decreasing the benefits of the API:s.

9.3 The Platform Leaders Ability to Enact Governance Activities

There are contextual factors in this SECO that limits Axis influence as platform leader and ability to enact certain governance activities. Axis does not raise entry barriers to the SECO, rather other factors are identified as keeping them high. The strongest factors influencing the entry barriers to the software ecosystem is the access to the end customers. To access the end customer in this SECO a third party developer most commonly need multiple relationships with vendors and other software manufacturers. Due to their two-tier business model Axis does not have direct channels to end customers and thus have limited ability to systematically facilitate governance activities related to them. The need of multiple relationships is enhanced by Axis not having a sales partner program for software and more specifically for ACAP applications. Furthermore, due to ACAP-companies often being relatively small it is up to them to adapt to other software and not the other way around, in order to access other company's customer base. This in combination with their two-tier business model incorporates several additional actors to the chain of distribution. Power is hence diffused over several different actors, such as distributors, resellers and system integrators, which are more or less connected to Axis and in the range of being affected by the governance activities exerted on the ACAP SECO.

9.4 Limited Variety

A majority of ACAP developers are active within the same industry. Additionally all ACAP developers in this study were active developers before starting to develop towards ACAP. This indicates that entry barriers are perceived as lower by developers already active within the industry, because they already have established connections and sales channels as well as knowledge about the business. The lack of developer variety imply that, in this SECO, the platform leader should not raise entry barriers, as suggested by Jansen et al (2009) but rather facilitate the entry of new participants which complement the ones already active. Young SECO:s commonly suffer from this problem with too few developers and low variety because there is not enough adopters of the underlying technology (Jansen et al., 2009). This creates an environment where there is difficult to get good returns on investment and hence there is an insufficient amount of new developers taking part in the SECO. Theoretically all camera customers are potential ACAP customers. But as mentioned in the previous section there are several factors restricting ACAP developers from accessing a large portion of these customers. Jansen et al (2009) propose several actions a platform leader can take in order to constrain this problem. Axis has adopted some of them when creating their API:s and partnership program. According to Jansen et al (2009) the next step would be to provide developers with financial incentives by sharing revenue or introducing direct rewards. As Axis are not yet conducting a large set of the governance activities presented in this study they should start by exploring these instead of providing financial support in order to create a sustainable SECO. The low variety may also be connected to the fact that the platform is implemented on a camera with limited features. Hence, the applicability of both the camera and the platform is limited.

9.5 Strategic conflict

Four out of six activities assessed as performed by Axis are not conducted with focus on ACAP: (1) The development program has a wide focus not providing sessions aimed at ACAP, (2) the ADP-program is focused to promote developers of high volume and broad applications rather than niche applications, such as ACAP applications. (3) Axis main marketing activities are conducted in order to increase network video solutions, i.e. camera sales and (4) the existing alliances were established prior to ACAP. Axis expressed strategy focus is on cameras and it is the authors perception that they want to use ACAP and its surrounding ecosystem as a supporting function in order to generate increased camera sales. Consequently, several activities does not have the, by theory, intended effect in this SECO. Additionally, some activities are prevented by choices Axis has made connected to their camera-strategy; their two-tier business model for selling cameras prevents third party developers to adopt certain business models, and ACAP developers are not invited to events due to focus being on companies having larger influence on cameras sales.

Furthermore, the absolute majority (figures hidden due to confidentiality) of Axis' business is today related to its products and features provided by them rather than the platform and ACAP applications. Hence, Axis' business model, partner program and overall strategy is focused on the products and not on ACAP or its SECO. Consequently this SECO is not prioritized and is not progressing in line with Axis' strategy of expanding its applicability. There are many activities performed with the camera strategy in mind and some of them obstruct the growth and development of ACAP. It is the authors' understanding that there currently is a conflict between Axis' overall product strategy and the strategy of the platform. This is supporting Bosch (2009) who stated that a platform leader of an application-centric ecosystem often struggle with a conflict between its product strategy and platform strategy.

10 Conclusion

The purpose of this thesis was attained by answering three research questions. These answers are presented in this section together with the overall results.

Research Question 1: Why do third party developers join and participate in Axis' software ecosystem?

End customer demand and a close relationship were found to be reasons for third party developers to participate in this SECO. The existing internal standard and the possibility of an external standard were further identified as benefits contributing to participation.

Research Question 2: **What makes third party developers hesitant towards joining Axis' software ecosystem?**

Axis two-tier business model creates the strongest hesitations toward joining and participating in the ACAP SECO, while doubts about the performance were the most commonly identified drawback.

Research Question 3: What governance activities are performed by Axis as a platform leader?

Axis performed six out of the 15 governance activities evaluated in this study. Seven of the remaining activities were assessed as not performed and 2 were assessed as partially performed. Many of performed activities were not performed with focus on ACAP but rather focused on facilitating camera sales.

Purpose: Explore how governance performed by a platform leader affects developer participation in a software ecosystem.

Unsystematically performed governance activities were found to increase the need of a personal relationship and good communication between the platform leader and third party developers in order for the latter to participate. Additionally, the creation and stabilization of API:s were found to be important for developer participation in a software ecosystem where the platform leader provide several product lines, supporting previous research by Bosch (2009).

As found in earlier research (Bosch, 2009), companies with application centric software ecosystems tend to have a conflict between their product and platform strategy. The findings in this thesis support that notion. Many governance activities are performed without the platform as a focal point, decreasing their potential positive effects on participation.

There are many contextual factors highly influencing participation, such as the platform leader's choice of business model. Even though the platform leader have large influence on the software ecosystem it does not necessarily mean that it has the largest influence or that it is large enough. By not selling directly to end customers the power is diffused over many vendors. This reduces the effects of governance activities and prevents enactment of others, which reduce the platform leader’s influence on participation.

10.1 Implications for Further Research

Software ecosystems as a field of research are relatively new. The few dominant authors in the field are calling out for more case studies in the area to formalize the field. The authors of this master’s thesis have provided an understanding of why developers join a specific ecosystem and formalized the context in the light of previous research.

The findings of this study highlight the importance of communication and relationships in new software ecosystems where governance is still unsystematic. As more hardware focused companies are looking to introduce SECO:s by creating a software platform the authors of this study deem it likely that they will face similar circumstances.

An interesting finding was how the contextual factors of the SECO affect the platform leader’s ability to control the entry barriers. By analyzing the ecosystem using the classification model practitioners and researchers can find the where the entry barriers are located and who controls them. It is the authors’ belief that platform leaders could be over confident in their power over their SECO.

Future researchers are encouraged to extend the validity of this study by making comparisons to similar ecosystems, made possible by the detailed classifications.

The possibility to improve the classification models is also provided. As more software ecosystems built upon hardware products are emerging the language is in need of evolvement.

The reader has been provided with an understanding of why developers join this software ecosystem and how the contextual factors have affected that decision. By conducting larger studies with multiple ecosystems the impact of contextual factors could be formalized.

Longitudal case studies, where changes in governance activities are studied over time and how it affects participation would create an even more fine-tuned understanding than what this thesis can provide.

11 References

ACCENTURE. 2009. *Developing Indirect Channels: A Structured Approach to Reaching New Customers and Growing Revenues* [Online]. Available: http://nstore.accenture.com/IM/MC/AccentureLibrary/data/pdf/developing_indirect_channels.pdf [Accessed 05/27 2013].

ADNER, R. 2004. A demand-based perspective on technology life cycles. *Advances in Strategic management,* 21**,** 25-43.

ANDERSSON, C. & RUNESON, P. 2007. A spiral process model for case studies on software quality monitoring - method and metrics. *Software Improve Pract,* 12**,** 125-140.

AXIS COMMUNICATIONS AB. 2009. Available: http://www.axis.com/corporate/press/industry_news/article.php?article=090921_applicationplatform.htm [Accessed 2301.05.14.]

AXIS COMMUNICATIONS AB. 2013. *Annual report 2013* [Online]. Available: http://www.axis.com/files/annual_reports/2012annual_eng.pdf [Accessed 04/12 2013].

AXIS COMMUNICATIONS AB 2011a. Axis Communications the driving force in network video - company presentation.

AXIS COMMUNICATIONS AB. 2011b. *Strengthen the position on the security market* [Online]. Available: Strengthen the position on the security market.

AXIS COMMUNICATIONS AB. 2013a. *About Axis Communications* [Online]. Available: http://www.axis.com/corporate/about/index.htm [Accessed 2013.02.26 2013].

AXIS COMMUNICATIONS AB. 2013b. *Applications ready to meet your needs* [Online]. Available: http://www.axis.com/products/video/compatible_applications/index.php [Accessed 2013.04.22 2013].

AXIS COMMUNICATIONS AB. 2013c. *Axis' Learning Center* [Online]. Available: http://www.axis.com/academy/ [Accessed 2013.05.04 2013].

BAARS, A. & JANSEN, S. 2012. A Framework for Software Ecosystem Governance. *In:* CUSUMANO, M., IYER, B. & VENKATRAMAN, N. (eds.) *Software Business.* Springer Berlin Heidelberg.

BARBOSA, O. & ALVES, C. A Systematic Mapping Study on Software Ecosystems. *In:* JANSEN, BOSCH, AHMED & CAMPELL, eds. Workshop on Software Ecosystems, 2011. 15-26.

BENBASAT, I., GOLDSTEIN, D. K. & MEAD, M. 1987. The Case Research Strategy in Studies of Information Systems. *MIS Quarterly,* 11**,** 369-386.

BJARNASON, E. W., K. REGNELL, B. 2012. Are you biting off more than you can chew? A case study on causes and effects of overscoping in large-scale software engineering. *Information and Software Technology 54,* 1107-1124.

BJÖRKLUND, M. P., ULF. 2003. *Seminarieboken: att skriva, presentera och opponera*, Studentlitteratur AB.

BONCHEK, M. & PAUL CHOUDARY, S. 2013. *Three Elements of a Successful Platform Strategy* [Online]. GBR Blog Network. Available: http://blogs.hbr.org/cs/2013/01/three_elements_of_a_successful_platform.html?utm_source=feedburner&utm_medium=feed&utm_campaign=Feed%3A+harvardbusiness+%28HBR.org%29 [Accessed 2013-03-18 2013].

BOSCH, J. 2009. From software product lines to software ecosystems. *Proceedings of the 13th International Software Product Line Conference.* San Francisco, California: Carnegie Mellon University.

BREM, A. & VOIGT, K.-I. 2009. Integration of market pull and technology push in the corporate front end and innovation management—Insights from the German software industry. *Technovation,* 29**,** 351-367.

BRYMAN, A. & BELL, E. 2011. *Business Research Questions,* New York, Oxford University Press Inc.

BUCKLIN, L. P. & SENGUPTA, S. 1993. Organizing Successful Co-Marketing Alliances. *Journal of Marketing,* 57**,** 32-46.

CECCAGNOLI, M., FORMAN, C., HUANG, P. & WU, D. J. 2012. COCREATION OF VALUE IN A PLATFORM ECOSYSTEM: THE CASE OF ENTERPRISE SOFTWARE. *MIS Quarterly,* 36**,** 263-290.

CHESBROUGH, H. W. 2003. *Open innovation: The new imperative for creating and profiting from technology*, Harvard Business Press.

CLARKE, T. & DELA RAMA, M. 2008. *Fundamentals of Corporate Governance*, Sage Publications.

COLLABNET 2004. Enabling Software Co-Development with Partners (White Paper).

CUSUMANO, M. 2004. The Business of Software: What Every Manager, Programmer and Entrepreneur Must Know to Succeed in Good Times and Bad. *Free press, NY*.

D'AVENI, R., MacMILLAN, I. 1994. Hypercompetition, Managing the Dynamics of Strategic Maneuvering. *Free Press, NY.*

DHUNGANA, D., GROHER, I., SCHLUDERMANN, E. & BIFFL, S. Software Ecosystems vs. Natural Ecosystems: Learning from the Ingenious Mind of Nature. Fourth European Conference on Software Architecture, 2010 Copenhagen. 96-102.

DIRECTORATE FOR FINANCIAL AND ENTERPRISE AFFAIRS 2006. Barriers to Entry.

DUBOIS, A. & GADDE, L.-E. 2002. Systematic combining: an abductive approach to case research. *Journal of Business Research,* 55**,** 553-560.

EISENHARDT, K. M. 1989. Building Theories from Case Study Research. *The Academy of Management Review,* 14**,** 532-550.

EKLUND, U. & BOSCH, J. 2012. Introducing Software Ecosystems for Mass-Produced Embedded Systems. *In:* CUSUMANO, M., IYER, B. & VENKATRAMAN, N. (eds.) *Software Business.* Springer Berlin Heidelberg.

FARRELL, J. & KLEMPERER, P. 2007. Chapter 31 Coordination and Lock-In: Competition with Switching Costs and Network Effects. *In:* ARMSTRONG, M. & PORTER, R. (eds.) *Handbook of Industrial Organization.* Elsevier.

GAWER, A. & CUSUMANO, M., A 2002. *Platform Leadership: How Intel, Microsoft, and Cisco Drive Industry Innovation*, Harvard Business Review Press.

GAWER, A. H., R. 2007. Platform Owner Entry and Innovation in Complementary Markets: Evidence from Intel. *Journal of Economics & Management Strategy,* 16**,** 1-34.

HAGEL, J. I., BROWN, J. S. & DAVISON, L. 2008. SHAPING STRATEGY in a World of Constant Disruption. *Harvard Business Review,* 86**,** 80-89.

HANSSEN, G., K 2012. A longitudinal case study of an emerging software ecosystem: Implications for practice and theory. *Journal of Systems and Software,* 85**,** 1455-1466.

HANSSEN, G., K & DYBÅ, T. Theoretical foundations of software ecosystems. IWSECO, 2012 Boston. 6-17.

HARTIGH, D., TOL, M. & VISSCHER, W. The Health Measurement of a Business Ecosystem. ECCON, 2006.

HUANG, P., FORMAN, C., CECCAGNOLI, M. & WU, D. J. 2009. WHEN DO ISVS JOIN A PLATFORM ECOSYSTEM? EVIDENCE FROM THE ENTERPRISE SOFTWARE INDUSTRY. *Thirtieth International Conference on Information Systems*.

HÄGG, I. & HEDLUND, G. 1978. *"Case Studies" in Social Science Research*, European Institute for Advanced Studies in Management.

HÖST, M. R., BJÖRN. RUNESON, PER. 2006. *Att genomföra examensarbete*, Studentlitteratur AB.

IANSITI, M. & LEVIEN, R. 2002. Keystones and Dominators: Framing operating and technology strategy in a business ecosystem.

IANSITI, M. & LEVIEN, R. 2004a. *The Keystone Advantage: What the New Dynamics of Business Ecosystems Mean for Strategy, Innovation, and Sustainability*, Harvard Business Press Books.

IANSITI, M. & LEVIEN, R. 2004b. Strategy as Ecology. *Harvard Business Review,* 82**,** 68-72.

IMS RESEARCH INSTITUTE 2012. Trends for 2013 - Video Surveillance Trends for the Year Ahead. IMS Research Institution.

JANSEN, S., BRINKKEMPER, S. & FINKELSTEIN, A. Business Network Management as a Survival Strategy: A Tale of Two Software Ecosystems. International Workshop on Software Ecosystems, 2009. 34-48.

JANSEN, S., BRINKKEMPER, S., SOUER, J. & LUINENBURG, L. 2011. Shades of gray: Opening up a software producing organization with the open software enterprise model. *The Journal of Systems and Software,* 85**,** 1495–1510.

JANSEN, S. & CUSUMANO, M. Defining Software Ecosystems: A Survey of Software Platforms and Business Network Governance. IWSECO, 2012.

KATO, K. & NEGORO, T. A theoretical review of network effects on platform products. Proceedings of the 51st Annual Meeting of the ISSS, 2007.

KIM, W. C. & MAUBORGNE, R. 1997. Value innovation. *Harvard Business Review*.

KITTLAUS, H.-B. & CLOUGH, P., N. 2009. *Software Product Management and Pricing*, Springer Berlin Heidelberg.

LAMM, J., BLOUNT, S., BOSTON, S., CAMM, M., CIRABISI, R., COOPER, N., DATSKOVSKY, G., FOX, C., KENNETH HANDAL, MCCRACKEN, W., MEYER, J., SCHEIL, H., SRULOWITZ, A. & ZANELLA, R. 2008. *Under Control: Governance Across the Enterprise*, APRESS.

LETHBRIDGE, T., SIM, S. & SINGER, J. 2005. Studying software engineers: data collection techniques for software field studies. *Empiric Software Eng.,* 10**,** 311–341.

LEXICON, F. T. 2013. *Governance* [Online]. THE FINANCIAL TIMES LTD 2013. Available: http://lexicon.ft.com/Term?term=governance [Accessed 2013.04.29 2013].

LIEBOWITZ, S. J. & MARGOLIS, S. E. 2001. *Network Externalities (Effects)* [Online]. Available: http://wwwpub.utdallas.edu/~liebowit/palgrave/network.html [Accessed 2013.04.23.

MANIKAS, K. & HANSEN, K. M. 2013. Software ecosystems – A systematic literature review. *Journal of Systems and Software,* 86**,** 1294-1306.

MOORE, J., F. 1993. Predators and Prey: A New Ecology of Competition. *Harvard Business Review*.

MOORE, J., F. 1997. *The death of competition: Leadership and strategy in the age of business ecosystems*, Harper Paperbacks.

NATIONALENCYKLOPEDIN 2012. *prestation.*

NIWOT RIDGE CONSULTING. 2001. *Risk Assessment: Risk Assessment Template for Software Development or Acquisition Projects* [Online]. Available: http://www.niwotridge.com/PDFs/RiskAssesmentTemplate.PDF [Accessed 2013.04.23 2013].

OECD. 2004. *OECD Principles of Corporate Governance* [Online]. Paris, France: OECD Publications Service. Avalable: http://www.oecd.org/corporate/ca/corporategovernanceprinciples/31557724.pdf [Accessed 2013-03-01 2013].

PCMAG.COM. 2012. *platform* [Online]. Available: http://www.pcmag.com/encyclopedia_term/0,2542,t=platform&i=49362,00.asp.

RAO, A. R. R., R. W. 1994. Brand Alliances as Signals of Product Quality. *Sloan Management Review,* 36:1**,** 87-97.

RESEARCH AND MARKETS 2012. Global Video Surveillance Market 2011-2015. Research and markets: Infiniti Research Limited.

ROCHET, J. C. & TIROLE, J. 2004. *Two-sided markets: An overview* [Online]. Available: http://web.mit.edu/14.271/www/rochet_tirole.pdf [Accessed 2013.04.19.

RUNESON, P., HOST, M., RAINER, A. & REGNELL, B. 2012. *Case study research in software engineering: guidelines and examples*, Wiley.

RUNESON, P. & HÖST, M. 2009. *Guidelines for conducting and reporting case study research in software engineering*.

SYDOW, J. & WINDELER, A. 1998. Organizing and evaluating interfirm networks: a structureationist perspective on network processes and effectiveness. *Organizational Science,* 27**,** 47-75.

TENG, B. S. & DAS, T. K. 2001. Trust, control and risk in Strategic Alliances: An Integrated Framework. Department of Management, Zicklin School of Business, Baruch College, City University of New York, USA.

VAN DEN BERK, I., JANSEN, S. & LUINENBURG, L. 2010. Software ecosystems: a software ecosystem strategy assessment model. *Proceedings of the Fourth European Conference on Software Architecture: Companion Volume.* Copenhagen, Denmark: ACM.

VAN DER SCHUUR, H., JANSEN, S. & 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, California: ACM.

VAN DER SCHUUR, H., SLINGER, J. & BRINKKEMPER, S. A Reference Framework for Utilization of Software Operation Knowledge. Software Engineering and Advanced Applications (SEAA), 2010 36th EUROMICRO Conference on, 1-3 Sept. 2010 2010. 245-254.

VAN GREMBERGEN, W. 2004. *Strategies for Information Technology Governance,* USA, Idea Group Publishing.

VILJAINEN, M. & KAUPPINEN, M. 2011. Software Ecosystems: A Set of Management Practices for Platform Integrators in the Telecom Industry. *In:* REGNELL, B., WEERD, I. & TROYER, O. (eds.) *Software Business.* Springer Berlin Heidelberg.

WILLIAMSON, O. E. 1991. Comparative Economic Organization: The Analysis of Discrete Structural Alternatives. *Administrative Science Quarterly,* 36**,** 269-296.

ZHU, F. & IANSITI, M. 2012. Entry into platform-based markets. *Strategic Management Journal,* 33**,** 88-106.

Appendix A: Tables

11.1 Interview instrument

11.2 Table 2: Reasons and benefits

Company	Company A	Company B		Company C	Company D		Company E	Company F
Interviewee	A1	B1	B2	C1	D1	D2	E1	F1
Reasons (Re)	Customer demand (Re1) Relationship (Re2) Open environment	Relationship (Re2) Geography	Customer demand (Re1)	N/A	Customer demand (Re1) External standard (Be2)	Future possibilities Marketing	N/A	Relationship (Re2) Internal standard (Be1) Low risk
Benefits (Be)	Internal standard (Be1) Piggyback on Axis	Internal standard (Be1)	Ease of installation	N/A	External standard (Be2)	Internal standard (Be1) Ease of installation Less infrastructure External standard (Be2)	External standard (Be2)	Scalability Less resources needed

Table 2 Reasons and benefits related to participation

11.3 Table 3: Statement about communication

Company	**Company A**	**Company B**		**Company C**	**Company D**		**Company E**	**Company F**
Interviewee	A1	B1	B2	C1	D1	D2	E1	F1
Statement about communication	Free communication Axis is a good partner	Close relationship	Very open communication	-	Good relationship Tight contact & support Fun	Very transparent and honest	Opaque communication Cannot find the right person to talk to Hard to get clear answers	Great communication Quick responses with good answers

Table 3 Statement about communication in relation to participation

11.4 Table 4: Hesitations and drawbacks

Company	**Company A**	**Company B**		**Company C**	**Company D**		**Company E**	**Company F**
Interviewee	A1	B1	B2	C1	D1	D2	E1	F1
Hesitations (He)	None	None	None	Performance (Dr1) Technical features Axis’ business model (He1) Unclear roles	Performance (Dr1)	-	Business model (He1) Lack of information about customers Strict rules of ADP	Performance (Dr1) Installment
Drawbacks	Performance (Dr1)	Performance (Dr1)	-	Performance (Dr1)	Performance (Dr1) Debugging	Performance (Dr1)	Information-gap	Performance (Dr1) No uniform camera compatibility

Table 4 Hesitations and drawbacks related to participation

11.5 Table 5: Interviewees at Axis

Role of interviewee
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

Table 5 Internal interviews at Axis

Appendix B: 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.

		t
1	**Introduction**	10
1.1	Explain that the material is not going to be shared with Axis without consent.	
1.2	Explain the purpose of the thesis.	
1.3	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.	
1.4	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
2.1	What is your role at the company today? Previous roles? (years per role)	
2.2	(If applicable) How much experience do you have from developing ACAP applications and analytics applications in general?	
2.2.1	In what roles have you acquired this experience?	
2.3	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**	
3.1	Tells us about your company? (number of employees, number of products, types of products)	
3.2	Tell us about your company's products (number of products, number of products to ACAP, number of products related to Axis)	
3.2.1	Is your product available as an ACAP application?	
3.2.1.1	(If applicable) Is your ACAP product also available as a server solution?	
4	**Reasons for developing**	

4.1	Why you /your company did chose (not) to start developing an ACAP application?	10
4.2	(If applicable) How did you come up with the application idea?	
4.3	What were the most important factors when choosing the Axis application platform? (Follow leads and dig deeper)	20
4.4	Was there anything that made you hesitate about developing for the Axis application platform? (Follow leads and dig deeper)	
4.5	(If company has chosen not to develop towards ACAP) What were the most important factors when choosing not to develop towards ACAP?	
4.5	What are the main areas of improvement for ACAP?	10

Appendix C: Survey Questions

1. What is your name and what role do you hold at your company? (open answer)
2. Which of the following statements do you perceive Axis to be doing today in regards to AXIS application platform (ACAP):
    a. Axis is expanding the application programming interface (API) of ACAP to enable you to access new functionality? (Yes/No/Don't know)
    b. Axis is improving the API to better utilize current hardware and / or software features? (Yes/No/Don't know
    c. Axis is improving the API based on partner feedback?
3. How do you receive information from end-users regarding how your ACAP product works in the field? (open answer)
4. Does Axis communicate identified end-customer needs to you? (Yes/No/Don't know)
    a. If yes: How does Axis communicate end-customer needs to you? (open answer)
5. Is Axis making new markets available for your ACAP application(s)? For example through facilitating new contacts, distribution channels etc. (open answer)
    a. If answer: In what way does Axis make new markets available to your ACAP application(s)?
6. Does Axis enable you to find relevant customers for your ACAP application(s)? (Yes/No/Don't know)
    a. If yes: In what way does Axis enable you to find relevant customers for your ACAP application(s)? (open answer)
7. Are you aware of Axis strategy for the Axis camera application platform (ACAP)? (Yes/No/Don't know)
    a. If yes: How was Axis strategy for the Axis camera application platform (ACAP) communicated to you? (open answer)
8. Please rate the APIs of ACAP in terms of overall performance. Taking for example speed, usability, stability, documentation and compatibility in to account
   (Very low (1) – very high (5) / no applicable)
9. Please rate the Axis application platform (ACAP) APIs in terms of ease of use: (Very hard (1) – very easy (5) / no applicable)
10. Please list, without any internal order, the five main advantages of the API of AXIS camera application platform (ACAP). (open answer)
11. Please list, without any internal order, the five main limitations or disadvantages of the API of AXIS camera application platform (ACAP). (open answer)
12. To what degree is development training with AXIS camera application platform (ACAP) available to you?
    (Very low (1) – very high (5) / no applicable/ Don't know)
13. To what degree is support for AXIS camera application platform (ACAP) available to you?
    (Very low (1) – very high (5) / no applicable/ Don't know)

14. To what degree do you perceive the AXIS camera application platform (ACAP) API to be stable? (very unstable (1) – very stable (5) / no applicable/ Don't know)
15. Is the AXIS camera application platform (ACAP) API changing more or less frequently than expected? (Less frequent/ more frequent / Don't know)
16. Do perceive Axis to facilitate financial support for development of ACAP applications? (Yes/To some degree/No/Don't know)
17. Do you perceive that Axis marketing activities benefits your ACAP product(s)? (Yes/No/Don't know)
    a. If yes: In what way do Axis marketing activities benefit your product(s)? (open answer)
18. Have you been in contact with other developers towards Axis camera application platform (ACAP)?
    (Not at all (1) – very frequent (5) / no applicable)
    a. If yes: How did you contact the other developer(s) and what was the reason of this contact for example technical support? (open answer)
19. Is there any reason to why you have not been in contact with other ACAP developers? (open answer)
20. Do you perceive Axis to be facilitating meetings between developers? (not at all (1) – very frequent (5) / no applicable)
21. Do you perceive Axis to be facilitating you in selling your ACAP product? For example by offering contact with their distributors etc.(Yes/No/Don't know)
    a. If yes: In what way do you perceive Axis to be helping you to sell your ACAP product? (open answer)
22. Do you perceive Axis to be helping you to distribute your ACAP product(s) (Yes/No/Don't know)
    a. If yes: In what way do you perceive Axis to be helping you to distribute your ACAP product?
23. Below you see a number of possible activities for Axis to perform as a platform leader, please rank these activities in accordance to their importance to you (options presented in randomized order)
    a. Facilitate distribution of my product
    b. Communicate customer need to partners
    c. Make new functionality available through ACAP
    d. Improve performance of camera API
    e. Communicate platform strategy
    f. Facilitate communication with other developers
    g. Facilitate sales of my product
    h. Improve ease of API use
    i. Keep API stable
    j. Make customers aware of partner products
    k. Provide development training
    l. Provide financial support to development