



UNIVERSITY OF GOTHENBURG

Software Asset Management in Large Scale Organizations

Exploring the Challenges and Benefits

Bachelor of Science Thesis Software Engineering and Management

Jody Swartz

Paulius Vysniauskas

University of Gothenburg
Chalmers University of Technology
Department of Computer Science and Engineering
Göteborg, Sweden, June 2013

The Author grants to Chalmers University of Technology and University of Gothenburg the non-exclusive right to publish the Work electronically and in a non-commercial purpose make it accessible on the Internet. The Author warrants that he/she is the author to the Work, and warrants that the Work does not contain text, pictures or other material that violates copyright law.

The Author shall, when transferring the rights of the Work to a third party (for example a publisher or a company), acknowledge the third party about this agreement. If the Author has signed a copyright agreement with a third party regarding the Work, the Author warrants hereby that he/she has obtained any necessary permission from this third party to let Chalmers University of Technology and University of Gothenburg store the Work electronically and make it accessible on the Internet.

Software Asset Management in Large Scale Organizations

Exploring the Challenges and Benefits

Jody Swartz
Paulius Vysniauskas

© Jody Swartz, June 2013.

© Paulius Vysniauskas, June 2013.

Examiner: Matthias Tichy

University of Gothenburg
Chalmers University of Technology
Department of Computer Science and Engineering
SE-412 96 Göteborg
Sweden
Telephone + 46 (0)31-772 1000

Department of Computer Science and Engineering
Göteborg, Sweden June 2013

Software Asset Management in Large Scale Organizations

Exploring the Challenges and Benefits

Jody Swartz

Department of Computer Science and Engineering
University of Gothenburg
Gothenburg, Sweden
jodys@webmail.co.za

Paulius Vysniauskas

Department of Computer Science and Engineering
University of Gothenburg
Gothenburg, Sweden
gusvysnpa@student.gu.se

Abstract— Software Asset Management (SAM) is a relatively new practice, which deals with efficient management of software assets within an organization. This practice is addressed more in the business aspect of organizations, especially in large-scale organizations. However, there is a lack of academic research that has been carried out in this field. The aim of this paper is to explore the challenges that large-scale organizations face when managing software assets and investigate the existing SAM tools, which could help to mitigate the upcoming challenges. A case study at Volvo IT Mainframe department was conducted and nine respondents were interviewed. The results included the identified challenges, the perceived benefits of SAM and a SAM tool comparison. The paper also provides suggestions for future research within this area.

Keywords— *software asset, software asset management, SAM, SAM tool, managing software assets.*

I. INTRODUCTION

Large-scale organizations, also known as enterprises, consist of several hundred or more employees and possess numerous different types of assets, such as buildings, equipment, or IT assets. The latter category comprises mainly of hardware or software within an enterprise. More specifically, software is most often treated as an intangible asset whereas equipment is a tangible asset (Bott, 2000). Moreover, software should be “treated like any other asset in the business”, such as hardware or property (Jakubicka, 2010).

However, many organizations undervalue their software assets and do not allocate enough resources on their proper management. Ben-Menachem (2007) states that methods to manage software assets are inadequate and emphasizes the low information transferability between managers inside an organization. Management is of vital importance to large-scale organizations, because they can strongly impact the success of an organization. The lack of management often brings up challenges, such as the increased IT costs or inability to identify what software is being used and where. It can even result in worldwide problems, such as the “Year

2000 problem (Y2K)” (Klint & Verhoef, 2001; Ben-Menachem, 2007). Y2K cost organizations huge amounts of money to fix because of their lack of software asset inventory databases.

The above reasons resulted in the need to formalize the activities for managing software. The ISO/IEC 19770(2012) group of standards was the first attempt to formalize the process of managing software in large organizations. It defined Software Asset Management (SAM) as a process framework, which enables an organization to prove that it is performing SAM to an adequate standard, to satisfy corporate governance requirements and ensure effective support for IT service management overall. Moreover, McCarthy and Herger (2011) identified the following benefits of adopting SAM:

- Cost savings from the recycled software licenses.
- Accurate forecast and planning data.
- Audit readiness.

SAM tools exist on the market to help organizations track their software assets. Microsoft (2006) states the following benefits of using SAM tools:

- The virtual nature of software assets requires organizations to rely on tools to conduct inventories and track software licenses.
- If organizations adopt the right SAM tools, they can improve the Return on Investment (RoI) on their software investments, and help eliminate errors from data collected manually.

Furthermore, not all SAM tools are cross-platform and this proves quite a challenge for e.g. mainframe environments. Organizations have a lot of critical data in mainframe environments, and existing SAM tools do not work well across different Operating Systems (OS). The mainframe environment is fairly different from regular computer networks, which are powered by e.g. Windows machines, because the mainframes can have multiple OS's installed on them. Also, their applications perform very

critical tasks and are usually more expensive to purchase and maintain.

There is a lack of literature, which focuses on SAM in large-scale companies, specifically those implementing mainframe systems. Therefore, this study aims to investigate the following Research Questions (RQs):

1. What challenges do large-scale organizations face when managing software assets?
2. How can the existing SAM tools be efficiently used in large-scale organizations to mitigate the forthcoming managerial challenges?

A qualitative research design was applied to answer the above RQs. Data of this study is based on the findings from interviews at the Mainframe department in Volvo IT, literature review and tool comparison.

The two main contributions of this paper are:

- Identify the challenges that the company faces when managing software assets.
- Research on existing SAM tools and find out how they can help to mitigate the forthcoming challenges and improve the lifecycle management process.

The study includes only SAM tools that are applicable for usage in large-scale organizations, specifically for the mainframe z/OS platform. In addition, this study focuses only on software assets specified by the ISO/IEC 19770 standard.

The paper is organized as follows: section I introduces the reader to the nature of SAM. Section II provides a technical background and explains the related concepts in more detail. Section III describes the methodology used in this study and explains the research process. The results of this study are presented in section IV and discussed in section V, where the solutions to the identified challenges are also proposed. Finally, section VI draws the conclusions of this study and provides possible guidelines for future work.

II. THEORETICAL BACKGROUND

In this section we provide existing theory that relates to the research presented in this paper, such as: what an organization defines as a software asset, the description of ISO/IEC 19770, how SAM can be implemented in an organization, followed by the background on SAM tools and the basic theory of the mainframe environment.

A. Software Assets

Bott (2000) identified in his study that software should be treated as a fixed asset. Fixed assets are assets that contribute to a company's productive capacity, directly or

indirectly. Furthermore, fixed assets are also known as tangible assets because of their physical existence. Klint and Verhoef (2001) and Ben-Menachem and Gelbard (2002) point out the importance of treating software as a tangible asset. However, Bott (2000) recognized that software is normally regarded as an intangible asset but only if it is purchased separately from the hardware they reside in. Most companies do not own the software they use, instead they own a license to use the software, because a license is more tangible (Bott, 2000). Software can have a high rate of change. It is dynamic, transitory, unstable and only partly deterministic (Ben-Menachem, 2007). Thus, software needs to be managed. Moreover, software assets record and retrieve knowledge about business processes, internal and external collaborations in the core of businesses and in relation with business partners (Sharifi et al., 2009). Our paper focuses on the software assets specified by the ISO/IEC19770-1 (2012).

B. Description of ISO/IEC 19770

ISO/IEC 19770 (2012) group of standards were developed to enable an organization to prove that it is performing Software Asset Management (SAM) to a standard, which is sufficient to satisfy corporate governance requirements and ensure effective support for IT service management overall.

The main goal of SAM is to ensure the software license compliance through employee education and established purchasing procedures, while minimizing software expenses (Holsing and Yen, 1999).

The ISO/IEC 19770-1(2012) categorizes software assets in 3 parts, which are as follows:

1. Software use rights, reflected by full ownership (as for in-house developed software) and licenses (as for most externally sourced software, whether commercial or open-source).
2. Software for use, which contains the intellectual property value of software (including original software provided by software manufacturers and developers, software builds and software as installed and executed).
3. Media holding copies for software use e.g. such as software which is installed on a device.

The ISO/IEC 19770(2012) group of standards currently consists of the following parts:

- 1) ISO/IEC 19770-1: There are 27 Processes which create a conceptual process framework for SAM. These processes are divided into 3 main categories:
 - *Organizational Management Processes for SAM* deal with the control environments for SAM, which establish and maintain the management system within the other processes, such as roles and responsibilities, policies, processes and procedures, corporate governance and the competence in SAM.

Furthermore, planning and implementation processes ensure appropriate preparation and planning for the effective and efficient accomplishment of SAM objectives.

- *Core SAM Processes* focus on the inventory processes for SAM, which ensure that the management objectives are being achieved, for example software asset identification, software asset inventory management and software asset control. The verification and compliance processes detect and manage all exceptions to policies, processes and procedures, including license use rights, for example software asset verification, software licensing compliance, software asset security compliance and conformance verification for SAM. The Operations Management processes and interfaces execute operational management functions that are essential to achieving overall SAM objectives and benefits, such as relationship and contract management, financial management for Sam, service level management for Sam, security management.
- *Primary Process Interfaces for SAM* focus on the life cycle process interfaces for SAM, which is to specify requirements for these life cycle processes such as change management process, acquisition process, software development process, software release management process, software deployment process, incident management process, problem management process and retirement process.

- 2) ISO/IEC 19770-2: Software identification tags - specifies SAM data, whereby software is tagged for identification and management.
- 3) ISO/IEC 19770-3: Software licensing entitlement tags - computer files, which provide identifying information for software licensing rights.
- 4) ISO/IEC 19770-5: still under development. It will define a common set of vocabulary for the ISO/IEC 19770 series.
- 5) ISO/IEC 19770-7: still under development. It will describe how tags should be managed.

C. Implementing SAM

This sub-section describes the approaches of implementing SAM within an organization, followed by the models for implementing SAM and a method of gathering Software Assets in a large-scale organization.

1) Four-tier system

To make it easier for an organization to implement SAM, the original ISO/IEC19770-1 was revised in 2012 and

a four-tier system was defined in terms of the outcomes of each tier to help an organization implement SAM in an incremental manner. The outcomes of each tier are:

- Tier 1: Focuses on gathering *trustworthy data* which are all relevant information about software assets within an organization.
- Tier 2: Covers the *practical management* such as the basic management of the control environment which include policies, roles and responsibilities.
- Tier 3: Focuses on the *operational integration* to improve the efficiency and effectiveness of SAM in an organization.
- Tier 4: Focuses on the *Full ISO/IEC SAM conformance*, which is integrated into the strategic planning of an organization.

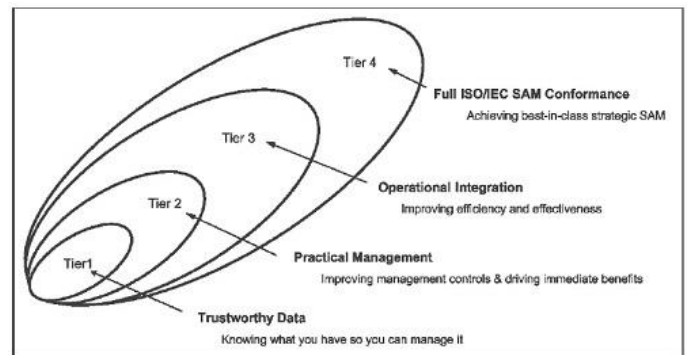


Figure 1. The four-tier system

Furthermore, the ISO/IEC19770 group of standards does not detail any processes in terms of approaches and strategies required to meet the requirements for the outcomes of a process.

2) Models

Holding and Yen (1999) proposed a software asset probation model and defined five problem areas, which drive the need for software management. They include ethical, legal, technical, managerial and economic issues. These 5 categories are used as motivation to implement SAM within an organization, based on the viewpoints from the key parties at stake: the end-user, the employer and the software publisher.

Ben-Menachem and Marliss (2004) described the “paradigm of change”. It is an integrated set of technologies, based on methods, tools and techniques for an appropriate overall IT inventory management. Thus, the “paradigm of change” accents that investment in creating and maintaining a software inventory is the first required step in proper long-term software management.

In addition to the “paradigm of change”, Ben-Menachem and Marliss (2005) defined a methodology software control by importance (SCIE) and exception. It allows optimization

of software items as assets based on an organization's software inventory and improve IT asset valuing to allocate development and evolution priorities.

McCarthy and Herger (2011) developed an IT service solution that integrated technology and data with process automation and business controls in support of Enterprise Software license management. The main aspects and outcomes of this solution are:

- *Discover software assets* by scanning distributed software license assets.
- *Reconcile purchased assets* through the reconciliation of procurement inventory.
- *Implement contract management* through the compliance with license terms & condition.
- *Produce business intelligence reporting* by having audit readiness and compliance.

3) SAM Inventory

Sharifi et al. (2009) suggested Configuration Management database (CMDB) as an approach when implementing a SAM inventory. CMDB is a repository for software assets of an organization, which provides an organized view of data and a means of examining data from different perspectives. Additionally, there are four different approaches in which to implement CMDB such as top-down, bottom-up, iterative and ad-hoc. Top-down approach is proposed to be the most effective as it is cheaper to implement, easy to use and less time consuming to implement.

D. SAM Tools

The existing SAM tools are delivered from vendors such as IBM, Microsoft and CA. Hence, this software is also offered for various platforms, including Linux/Unix, Windows and Mac machines and mainframe computers running z/OS operating systems.

Typically most of the SAM tools operate by scanning the present network (Jakubicka, 2010). During the scan they are able to identify installed products and retrieve related information, such as product name, size and version number. Later all this information is stored in a data file on a desired machine.

E. Mainframe Environment

Mainframes are a large type of server, which plays a central role in the daily operations of most of the world's largest corporations. They are primarily used for bulk processing, such as enterprise resource planning and transaction processing (Ebbers et al., 2011).

There are five OS's, which dominate mainframe system usage: z/VM, z/VSE, Linux for z series Z/TPF and z/OS. These OS's were developed by IBM. Each of them have different characteristics and purposes, which could individually reside on a logical partition (LPAR) and run simultaneously on a single mainframe computer (Ebbers et

al., 2011). Furthermore, Ebbers et al. (2011) state that the z/OS is the most widely used mainframe OS, because it is stable, secure, continuously available, and scalable environment for applications running on the mainframe.

According to Ebbers et al. (2011), the roles and responsibilities in a mainframe department of a large-scale organization are wide, varied and range from end-user to system administrator. It takes skilled staff to keep mainframe computers running smoothly and reliably.

III. RESEARCH APPROACH

We selected the qualitative approach (Creswell, 2009) and performed an exploratory case study (Robson, 2002) at Volvo IT, combined with literature review (Kitchenham, 2007). An exploratory case study centers on understanding the situation and finding out the insights within the contributors (Robson, 2002). The choice was influenced by the need of qualitative data in order to investigate the current situation of SAM within large-scale organizations. By interviewing the Mainframe department staff involved in the software asset lifecycle, we were able to investigate the challenges they face and what they would like to see improved. We then validated our interview review results against literature results, thus strengthening our research findings. Moreover, the research of market for SAM tools resulted in a comparison of several available products and their functionalities. The result would be used to recommend a viable tool for Volvo IT.

A. Research Setting

The case study was carried out in cooperation with a global IT company, Volvo IT (part of Volvo Group). More specifically, we worked in four divisions within the Mainframe department, located in Gothenburg, Sweden. The Mainframe department provides customers with services, which include installation, maintenance and administration of hardware and software. Hence, the department uses IBM Mainframes. The department was interested in refining its SAM processes and a SAM tool which would meet their needs. While staying at the site, information was gathered about the SAM tools used and the software asset related details (licenses, cost). Furthermore, access was given to their internal documents, such as current life cycle definition. Apart from the financial dimensions, legislative and operational aspects of asset management (Jakubicka, 2010) were observed, by discussing and interacting with software coordinator and product managers.

B. Research Process

The research was undertaken in three stages. The first stage was comprised of reviewing the available literature, necessary for the creation of a theoretical background in SAM. This data was used to identify the possible challenges in managing software assets, which would be reflected in the interview guide. Identifying available SAM tools was also required before meeting the company representatives.

The second stage consisted of creating the interview questionnaire, a presentation of the research project to hand out to the interviewees and conducting the actual interviews. The duration of this stage was carried out at the research site and we gathered the essential information for the third stage.

The third stage focused on the synthesis of our findings by comparing the interview results to the literature review results, as well as researching on whether there are any viable SAM tools that would fulfill the respondents' needs we discovered.

C. Data Collection

1) Literature Review

Besides the collaboration with the company, a systematic literature review (Kitchenham, 2007) was conducted. The reason for selecting the systematic literature review was that we needed to systematically summarize all existing information about the challenges and benefits of SAM as well as support our technical background. This information was gathered from available scientific databases.

The scientific databases were used include:

- IEEE Xplore
- ACM Digital Library
- Chalmers Library
- Science Direct

As outlined by Booth et al. (2003), sources can be categorized into three different kinds. We have used this method to classify our sources:

- Primary Sources: Raw data, i.e. interviews.
- Secondary Sources: Research reports i.e. academic articles, academic journals and conference papers.
- Tertiary Sources: Books and articles i.e. organizational papers, magazines, newspaper.

The research papers and organizational papers published between 1999 and 2013 were included as the secondary resource for this study. The reason for selecting this time period was that in 1999 large-scale organizations were faced with Y2K, which led to organizations trying to identify and define what software assets are.

As it is important to devise and follow a search strategy (Kitchenham, 2007), the following search terms were used: "software asset management", "SAM", "software asset", "managing software assets", "SAM tool".

As exclusive criteria, we excluded papers which specifically focused on the ISO/IEC 20000 standard for Information Technology Service Management (ITSM), as it focuses predominantly on IT assets rather than specifically on software assets.

As this area has not been widely researched at an academic level, we considered organizational publications as valuable information sources too. Furthermore, this research is limited to papers only published in English.

2) Interviews

Semi-structured interviews were the main source for collecting the required data. Thus, an interview guide was designed containing eight questions, of which the majority were open-ended questions (see Appendix A). The questionnaire was divided into three sections and designed in such a way, that after having asked the respondents specific questions, a small presentation was conducted to introduce the topic of SAM. The interview then continued, where the initial questions were revisited. This enabled the respondents to give their insights on the same matter, but from the SAM perspective. Thus, allowing an analysis on how respondents perceive the SAM concept and whether they see any benefits in it.

Hence, nine face-to-face interviews were conducted using the previously described interview guide. The interviewees have the following responsibilities: three Product Managers, two Mainframe Technicians, two Team Leaders, a Global Purchaser and a Software Coordinator. In addition, all of the interviews were recorded and later transcribed in order to ensure the validity and accuracy of the findings. The identities of the interviewees shall not be revealed.

D. Data Analysis

The collected data was analyzed using the thematic analysis method. We selected it due to its flexibility and ability to minimally organize and describe the data in rich detail (Braun and Clarke, 2006). Moreover, the analysis was conducted according to the 6-step guide as described by Braun and Clarke (2006):

- 1) Familiarizing with the data. We gathered relevant articles, and conducted the literature review. Then we conducted the interviews and transcribed them.
- 2) Generating initial codes. We looked through the ready transcripts to highlight the important ideas (codes).
- 3) Searching for themes. This stage included grouping the codes into themes and finding connections between interviews and literature.
- 4) Reviewing themes. We reviewed the identified themes to see if they are not too broad and if the data extracted for each theme are coherent with the theme itself.
- 5) Defining and naming themes. During this step we named the themes depending on the most important issues they described.

- 6) Producing the report. We summarized all the findings from the literature review and interviews, which were extracted throughout the analysis stages described above.

This means that we started from coding and reviewed the findings several times to extract the more important concepts, which could become themes. Next, we tried to find codes that would be common both in the interview data and literature. This resulted in several themes that were named according to the issue they addressed.

E. Data Validity

Since this study is mainly based on interviews, it was important to ensure data validity. This was achieved with the help of:

- Supporting literature. A review of related academic publications allowed us to validate our interview findings against those described in high quality academic papers.
- Company’s own documents. We were provided with several reports, explaining the present software lifecycle process within the company. This gave us the possibility to see whether our respondents really understood and used the processes described.
- Investigative triangulation. We carried out interviews with respondents who have different roles and responsibilities within the company. This allowed us to gain confirmation of our findings through the convergence of different perspectives.

Moreover, when comparing SAM tools, we relied on the information on publishers’ websites. There was a difference in the amount and quality of the information provided on the websites. Therefore small discrepancies between the described SAM tool functionalities might be present in the results table.

F. Limitations

At least three limitations were distinguished. The first one was the lack of relevant academic resources on software asset management (SAM). This field has not yet received much research, however it is being questioned in the industry and business. Therefore, this research paper expanded the range of searching from scientific databases to industrial publications and company reports. However, due to our choice of search terms and strategy, we might have missed including some relevant papers.

Another limitation was the necessity to focus only on mainframe environments, as the correspondents from our collaborative company work in this environment only. Considering the fact that there is even less academic research on SAM within mainframes. In general, this study was mainly restrained because of the limited number of relevant academic publications.

Furthermore, the selection of SAM tools to compare was delimited by selecting only the ones available for the mainframe z/OS operating system. This is because our interviewees work entirely in the mainframe environment. Another limitation was that we could not try these tools in real environment. This was due to time constraints and also the tools are commercial software, and require the purchase of their respective licenses.

IV. RESULTS

This section presents the findings of this study and is divided into three parts: literature review, interviews and SAM tool comparison.

A. Literature Review

This section reviews the current known challenges, faced when managing the software assets, and the benefits of SAM.

1) Challenges when Managing Software Assets

The challenges faced when managing software assets listed below can be divided into 3 areas: Tools challenges (see Table 1), Managerial challenges (see Table 2) and Organization challenges (see Table 3).

a) Tool Challenges

Tools Challenges	References
Lack of software assets information in inventory	Klint and Verhoef, 2001
Lack of centralized software asset repository	Ben-Menachem and Marliss, 2004
Lack of tools leads to insufficient software asset utilization	McCarthy and Herger, 2010

Table 1. Tools challenges

The *lack of software asset tools* is identified by many authors (Klint and Verhoef, 2001; Ben-Menachem and Marliss, 2004; Sharifi et al., 2009; McCarthy and Herger, 2010) as a main challenge when managing software assets.

Klint and Verhoef (2001) identified in their study that the *lack of inventory information* prohibits organizations to have insight in their total IT spending. According to Ben-Menachem and Marliss (2004), most organizations lack a central repository concerning what files exist, their locations, the relationships or dependencies existing between them, their ‘owners’, their birthdates, and modifications and their growth or decay rates. Ben-Menachem (2004) pointed out that visits to tens of major sites around the world indicated that the common situation is either no inventory or a primitive, out of date inventory. Furthermore, Ben-Menachem (2007) identified that the most sophisticated management seen at a major installation was an Excel spreadsheet table for systems/programs inventory

at a major insurer. McCarthy and Herger (2010) identified in their study that because of the lack of tools to measure and monitor both usage and availability of these licenses have made it *difficult to measure software asset utilization*. Moreover, this causes complications with compliance regulations.

b) *Management Challenges*

Management Challenges	References
Information transferability	Ben-Menachem, 2007
Insufficient documentation	Ben-Menachem and Gelbard 2002

Table 2. Management challenges

Ben-Menachem (2007) states that the basic problem of SAM is *information transferability*. Moreover, management needs must include information similar to the requirements for all expensive assets. Similarly, Ben-Menachem and Gelbard (2002) concluded that CFOs and CEOs lose patience with their IT department’s *inability to document* and justify their expenses, this is due to a lack of management processes.

c) *Organizational Challenges*

Organizational Challenges	References
Pressure of managing software systems in an organization	Sharifi et al. 2009
Software running in organization	Sharifi et al. 2009
Lack of interdepartmental data sharing	Ben-Menachem and Gelbard, 2002

Table 3. Organizational challenges

Sharifi et al. (2009) identified in their study that organizations are under the *pressures* of managing software systems, which are bigger and more complex as well as meet the *increasing demands for higher quality* to meet organization’s objectives. Also, Sharifi et al. (2009) state that organizations do not know how many software is running within their organization, which leads to inconsistencies with counting the software assets in an organization.

All managers need *access to the portion of the budget* – capital costs, services and staff time-consumed by each component and service in the enterprise environment. Lacking this information, the organization risks leaking precious financial resources through assets with inordinately

high life-cycle costs in ways that do not reflect corporate priorities. They are unable to replicate best practices through the organization because they cannot pinpoint groups successfully more with less (Ben-Menachem and Gelbard, 2002).

2) *Benefits of SAM*

According to Holsing and Yen (1999) implementing SAM can aid in avoiding legal ramifications as well as increase employee productivity. Furthermore, the ISO/IEC 19770-1 (2012) describes the overall benefits when good practice in SAM is achieved, which are risk management, cost control and a competitive advantage.

B. *Interviews*

In order to collect data on SAM challenges and benefits, nine face-to-face interviews were conducted at Volvo IT. Interview questions can be seen in Appendix A. The interview results are grouped into three themes: challenges when managing software assets (Table 5), perceived benefits of implementing SAM (Table 6) and suggestions for SAM tool functionalities.

Abbreviation	Description
PM	Product manager
MT	Mainframe technician
TL	Team leader
GP	Global purchaser
SC	Software coordinator

Table 4. Abbreviation meaning

1) *Challenges*

Challenge	References
Unclear lifecycle	PM
Unclear responsibilities	SC, TL
Tracking software	TL, MT, GP, PM, SC
Excel sheet management	PM
Dependencies/upgrade	TL
Redundant software	MT
Money loss	GP, TL

Table 5. Identified challenges

The interviewees identified a number of challenges, which include lifecycle management, software tracking, redundant software and costs:

a) *Lifecycle Management*

Our investigation showed that the Mainframe department at Volvo IT carries out some lifecycle management activities despite not having the overall process clearly defined. This became obvious both after conducting the interviews and analyzing the documents provided by the company. One of the product managers (PM) stated that for

the lifecycle planning they use information from IBM, one of their biggest vendors. The PM added that:

“All big software, at least, in Volvo IT should have a product lifecycle plan.”

One of the core roles in the department for practical lifecycle management is the Software Coordinator (SC). The SC communicates with the purchase department to order new software. However, there are still people sharing the same responsibilities, which is not good because of the extra confusion. One of the team leaders (TL) complained about the lifecycle and dividing responsibilities:

“It is an own process and we have some unclear responsibilities, e.g. is it my team’s responsibility, or is it another team’s responsibility when it comes to lifecycle management.”

b) Tracking the Software

The difficulties when trying to track the software usage was the challenge that our interviewees emphasized the most. The SC shared that:

“It is not easy for a product manager to keep track of software.”

Mainly the respondents talked about the inability for them to gather full data on every external customer, buying their services, i.e. what software the customer runs, how often it is used. One of the interviewees, the Global Purchaser (GP), admitted that:

“The challenge we have is to know how much we use the software, what customers are using the software, do we really need it.”

The mainframe has a lot of customers, using different products, so it is very important to keep information about programs in use and related information, such as releases and upgrades.

Moreover, several respondents complained about the enormous Excel sheets that currently are being used in order to store information about customers’ software. One of the PMs claimed that:

“... just now we use Excel spreadsheets and we know now for sure that they are not matched and not correct”

Another respondent, a TL, mapped software tracking to the upgrade issues. The TL mentioned the need to know the dependencies of every software product on other products:

“...so if you upgrade software B, what do you have to do with software C, D and E to be compatible with the new release”.

c) Redundant Software

The inability to fully track the software usage can lead to redundant usage of programs, which consume valuable storage and financial resources. This situation can emerge because of several reasons. One of the Mainframe Technicians (MT) mentioned the difficulty when identifying the program owner:

“...I need to find them [users] and many times you just keep things because you can’t find the users”

Moreover, the MT brings up licensing issues as a cause for redundancy:

“Because some software have keys and you can’t use it without keys, some have not and since Mainframe has been around for very many years, it can be that people use things that they shouldn’t use, but it still works, because the systems are usually backwards compatible so you can still run programs from the 70s-80s.”

d) Costs

As mentioned before, redundant software being run can impact the total costs by increasing them exponentially. Usually every software item requires an annual license renewal and often unnecessary product licenses are prolonged. This is very significant in the studied department, as mainframe software is very expensive. The GP told us that software is the single biggest item on the Mainframe and it makes 40 percent of the total budget. The respondent also added:

“That is why it’s crucial to keep track of software we have, and that it’s really used so we get a benefit for all this money.”

2) Perceived Benefits of SAM

Perceived benefit	References
Optimization	PM, TL, MT, GP
Work quality	TL
Clear responsibilities	PM, MT
Better vendor relations	PM
Reduced costs	PM, TL, GP, MT

Table 6. Perceived benefits

After being introduced to the concept of SAM, our respondents identified several benefits, which they thought they would experience by implementing SAM. These benefits include: infrastructure optimization, increased clarity of managerial activities and reduced costs for software.

a) Optimization

One of the ideas of SAM is infrastructure optimization. Good understanding of the assets an organization possesses, allows the removal of redundant software assets and optimizing the inventory to a high level. Most of the interviewees agreed on this benefit. One of the PMs stated:

“And of course, if you have control, you can avoid situations where you have a lot of software doing the same thing. So you can optimize that.”

Another respondent even considered optimization as a quality issue, which impacts the work environment and things such as reduced working hours. Moreover, the GP explained that they already practiced this activity to some extent:

“...is there anything that should be thrown out and then we give the vendor... this is the list for the next 3 years, this is the software that we want and for these volumes.”

Therefore optimization of assets tends to be an important SAM benefit for the organization studied.

b) Clarity of Management Activities

Optimization and therefore better control over software assets certainly adds more clarity to the management activities. A few of our respondents saw clarity as a very possible benefit when adopting SAM. One PM explained that:

“We all would benefit by having fewer questions and fewer discussions <...>. Everything would be clearer for everyone who needs to know about the current status.”

The respondent mentioned an unforeseen extra benefit that would be implied by increased clarity:

“Another benefit would be that you can avoid problems with audits by vendors.”

So increased clarity would not only make the management easier within the organization, but would also improve relations between the organization and its vendors.

c) Reduced Costs

Increased clarity and optimized software assets indicate the absence of redundant products. This impacts on the total costs for software, which in this case can be significantly decreased. Several of the interviewees agreed that SAM practice would help the company to reduce costs. As mentioned by the GP before, mainframe software is very expensive and plays a big integral role in the total budget. Therefore, these findings allow us to consider cost reduction as one of the biggest benefits for the studied organization.

3) Suggestions for SAM Tool Functionality

One of the questions the respondents were asked was what they expect from a good SAM tool or what features they miss from the tool they currently use. Almost all of our interviewees have been in contact with SAM tools, particularly “Tivoli Asset Discovery for z/OS” from IBM and a few of them use it in on a daily basis. We received answers, which we grouped into two categories:

a) Additional Details

The interviewees, who had some experience with SAM tools, emphasized the wish to be provided with more information and more flexibility. They would like to see additional details in the inventory, such as release information, upgrade availability, license expiration, owners of the product etc. One of the respondents mentioned that financial information should be visible only for authorized personnel. Thus, privacy issues should be considered. Another respondent thought about categorization:

“I guess we would like to have some internal classification of the software, I mean, is this class A product or class B product or something. Some way of classification that this is a strategic software that we have...”

A few of the interviewees raised up the requirement to add information manually to a SAM tool or to feed a database with the information from a SAM tool. However, adding the data manually is not recommended. Even though it is technically possible, it would become difficult to maintain SAM software during version changes. One of the PMs claimed that:

“We should have a tool which collects information from the servers and then load into the database, then you could use excel spreadsheets, pdfs.”

Yet the PM was not sure about what particular information should be kept within the database:

“What type of data we could get <...> perhaps is when the license expires, ordering the next release and getting ready with the migration process. The important question here is what valid information we need for the products in the database.”

Moreover, another interviewee complained that often it is difficult to interpret the data from a SAM tool and it requires help from skilled specialists. Thus, there should be more details provided in reports from SAM tools but at the same time it should be kept very simple in order to make it understandable for staff with less knowledge in the field. So our described findings leave a lot of space for improvements for SAM tools.

b) Accuracy

Another issue for improvement that we discovered was the accuracy of SAM tools when detecting running software. Despite the fact that all of the interviewees were

	Tivoli Asset Discovery	P-Tracker	BMC Atrium Discovery	XBridge DataSniff	Tibco Mainframe Service Tracker
Automated reports	Yes	Yes	Yes	Yes	Yes
Usage monitoring	Yes	Yes	N/A	N/A	Yes
Product title/vendor info	Yes	Yes	Yes	Yes	Yes
User info	Yes	Yes	Yes	Yes	Yes
Product location tagging	Yes	Yes	Yes	Yes	Yes
License/support information	Yes	Yes	Yes	Yes	Yes
In house application tagging	Yes	Yes	No	No	Yes
Version identification	Yes	Yes	Yes	Yes	Yes
Consolidation suggestions	Yes	Yes	Yes	No	Yes
CICS transaction monitoring	Yes	Yes	No	N/A	No
Multiplatform support	No	No	Yes	Yes	No
Cloud support	No	No	Yes	No	No
Various dataset formats	No	No	No	Yes	No

Table 7. Comparison of SAM Tools for z/OS

positive about these tools, a few of them mentioned problems with the results they provide. The SC remembered trying to use a previous SAM tool:

“One time when I talked to a vendor and he asked how many LPARs were being used and that tool said three, and I reported it to the vendor. Then we discovered from other sources that it was actually six more LPARs. It was quite embarrassing.”

One of the MTs even mentioned accuracy issues with the currently used “Tivoli Asset Discovery for z/OS”:

“This product works in a way that it recognizes different modules that are used and I have currently, today reported 3 problems to IBM, because it does not recognize some of these modules.”

Furthermore, the GP stated:

“You need to be able to trust the result, because first of all you need to make sure that the tool

finds everything that it can actually detect all the software that is installed, and secondly you need to make sure that it detects them correctly.”

So the result accuracy issues provide a lot of space for future improvements too.

C. SAM Tool Comparison

In addition to the interviews and literature review, we performed a comparison of available SAM tools (see *Table 7*). It is based on the suggestions for SAM tools that were extracted from the interviews as well as feature lists, provided by publishers. The intention is to find out whether there exists a SAM tool for z/OS platform that can fulfill most of the requirements, provided by the interviewees.

Looking at the table outline, it is clear that all the tools have similar basic functionalities. They scan the network, identify running products and provide additional information, such as program title, owner, place where it is run and license/support related information. Most of the

tools provide usage statistics and can suggest which modules are redundant, so they can help with optimization.

Regarding the challenges, identified during interviews, it is clear that SAM tools can help to mitigate these challenges. Having a tool, which can identify running products and create an inventory, facilitates the process of software tracking and usage monitoring as well as maintenance/upgrade issues. The full overview of inventory helps to eliminate redundant software, which also can result in reduced costs. Moreover, the possibility to create automated inventory would allow to drop the inefficient Excel table usage to keep track of software. However, the management challenges (unclear responsibilities and lifecycle process), which we discovered during interviews at the particular organization, require to look at the origins and cannot be solved by just usage of SAM tools. This is discussed in further detail in the following section.

Finally, the “Tivoli Asset Discovery for z/OS”, currently used at Volvo IT, Mainframe department, is a good choice as it supports the majority of the required functionalities. However, if Volvo IT, Mainframe department would like to have a cross-platform tool, they would have to look at “BMC Atrium Discovery” or “XBridge DataSniff”. Moreover, only the product from BMC has support for cloud computing, which is becoming a trend in the IT world.

V. DISCUSSION

In this section we discuss the interview and literature findings on challenges when managing software assets as well as benefits of SAM. Further, we discuss the SAM situation at Volvo IT, Mainframe department, and propose an approach for improving the lifecycle management process. Moreover, we conclude on the SAM tools we have reviewed.

A. Challenges

Our study revealed that there were not many identified challenges of managing software assets in the existing literature as we expected. Thus, we were able to gather more varied information on challenges from our interviews at Volvo IT.

One common challenge was the information availability. Several authors, including Klint and Verhoef (2001), Sharifi et al. (2009), McCarthy and Herger (2010) have mentioned it in their works. This was also one of the most accented issues during our interviews. Poor information about the inventory within an organization causes additional challenges, such as managerial difficulties and increased money loss. The financial aspect is a very important issue, because mainframe software is extremely expensive. In organizations, like Volvo IT, it makes almost a half of the total budget, so it is essential to keep the costs to a minimum.

The lack of information is mostly caused by the absence of SAM tools. Ben-Menachem (2007) states that during

visits at a major insurance company, he discovered that the way they tracked inventory was through the use of Excel tables. Surprisingly, the Mainframe department at Volvo IT also tracks inventory through the use of Excel sheets, which are used along with a SAM tool. Despite the employees’ complaints about the Excel sheets, which make the management process inefficient and complicated, these tables play an important role within our studied organization.

Another interesting challenge that we discovered during interviews was redundant software. In the mainframe environment it is common to have a lot of programs running, but sometimes there are modules, which run secretly, because they could not be identified. It is possible that some software is kept running intentionally, because it is unclear if it is really owned and used by someone or not. All these problems are caused by the inability to monitor software, its usage and keep required details. Moreover, the lack of the information on software dependencies can make it very complicated to maintain software and this causes problems during upgrades.

Summing up the challenges, it seems that the main problem is the lack of information about inventory. This is the reason, which likely causes the rest of the challenges. The lack of information implies on poor management of the assets, which then results in increased costs, inefficient usage and even redundant software. Thus, it is important to have a clear lifecycle process for software assets, which would be compliant with the ISO/IEC 19770 (2012) standard of SAM. Furthermore, a proper SAM tool should be taken into consideration in order to keep track of software assets.

B. Benefits

Similarly to exploring the challenges, our results revealed that there were very little information on the benefits identified in the existing academic literature. The ISO/IEC 19770-1 (2012) standard defined risk management, cost control and competitive advantage as the overall benefits of implementing SAM. It remained the main academic source. Benefits tend to be a more likely an industry related subject, which strongly depends on the individual organization. Hence, we were able to identify several benefits that employees at Volvo IT Mainframe perceived after being introduced to SAM. All of the perceived benefits are compliant with the ISO/IEC 19770-1 (2012) standard.

One of the identified benefits was optimization. It could be related to the competitive advantage, defined in the ISO standard for SAM. Optimized infrastructure can be defined as, there are no redundant software running and only the necessary modules with the required number of licenses are kept. It also maximizes the end-user productivity. Almost half of our respondents identified it as a potential win. Considering this and the opinions of several interviewees in regards to redundant software they currently have, it became

clear that our studied organization would experience this benefit if SAM was implemented.

Next, our interviews showed that an increase of clarity between the roles and their responsibilities in asset management is required. Currently, our studied organization has some dedicated roles, such as the Software Coordinator and Global Purchaser. However, some of the product managers and team leaders share unclear responsibilities in regards to the software lifecycle. Their opinions confirm that the organization needs clearer definitions for the management of every software asset's lifecycle, which would provide increased clarity. Thus, improving work efficiency and productivity.

Furthermore, the benefit of reduced costs was accented both in the ISO/IEC 19770-1 (2012) standard and among our respondents. Despite the fact that our studied organization does not experience much financial loss, our interviewees perceived cost reduction as a probable benefit. Knowing that they have a particular amount of redundant software, introducing SAM would help to cut additional costs. Moreover, SAM introduction in the studied organization would serve as prevention against eventual increase of total software spending.

C. SAM Approach Proposal

The results from the interviews indicated that the mainframe department of Volvo IT carries out some lifecycle management processes but it has not been clearly defined. This can lead to an increase in pressures for meeting business objectives by departments within an organization and poor management of the software assets (Sharifi et al., 2009).

To address the previously mentioned challenges, we would suggest the Mainframe department at Volvo IT to draw up a SAM plan which handles the people, processes, products/technology and vendors involved with SAM within the department. According to Holsing and Yen (1999), with an effective and enforceable SAM plan, organizations can avoid legal ramifications and provide potential opportunities for increased employee productivity.

The basis of this SAM plan could be achieved through the use of the four-tier system, specified in the ISO/IEC 19770-1 (2012) standard. Most organizations are unbeknown to the fact, that they already perform some of the ISO/IEC 19770-1 (2012) processes because it is naturally a necessity in order to acquire and use software assets. The four-tier system could be used as a guideline/checklist for implementing SAM within the studied department. Moreover, the fact that the Mainframe department currently lacks a well-defined lifecycle process for its products, it is important to start with creating a good inventory and defining managerial activities, which would be achieved after implementing first and second tiers. Third and fourth tiers are not necessary to focus on in the beginning and they could be implemented later once SAM is

defined in the department. Therefore, we advise that the Mainframe department should focus on the first and second tiers. However, the ISO/IEC 19770-1 (2012) itself, only outlines all the processes and outcomes for ISO/IEC 19770-1 (2012) compliance. Furthermore, it does not specify strategies or approaches for reaching the outcomes of those processes within the respective tiers. There are many different approaches that the department can choose from, such as: the software asset probation model (Holding and Yen, 1999), paradigm of change (Ben-Menachem and Marliiss, 2004), IT service solution (McCarthy and Herger, 2011) or even some commercially popular approaches which are out of this papers scope. Thus, an organization has the freedom to specify their own specific strategy or approach for compliance with the ISO/IEC 19770.

Regarding the first tier of the ISO/IEC 19770-1 standard, we would advise Volvo IT to focus on gathering all known information of their software assets within their department through the use of a SAM tool (see section D). In order to populate the database with software asset data, we would propose the use of CMDB's top-down approach (Sharifi et al., 2009). Another method of calculating which could aid in defining the most valued software asset, would be to implement Ben-Menachem and Marliiss (2005) methodology software control by importance (SCIE) and exception. This method allows consolidation of all information about the mainframe software assets and respective licenses in one place.

As for the second tier, which focuses on the practical management, we would suggest to invest more in formal education. The interviews revealed that some of the respondents did not feel comfortable when using the SAM tools because of not having the required skills. Therefore employees, especially those with product manager and team leader responsibilities, should be educated and encouraged to get more involved in the management process. Furthermore, additional resources should be allocated on defining the proper lifecycle process for every software. It is important that each and every software asset is managed from its procurement to its retirement. Despite the fact that the majority of company's vendors have adopted the ISO/IEC19770-1 (2012) standard, much individual input is required in order to achieve the conformance within the organization. Thus, clearly defining software lifecycle management and implementing a SAM strategy would be a good improvement for the studied department.

D. SAM Tool Selection

The research on SAM tools, available for the z/OS platform, resulted in a comparison table of five products from different vendors (see Table 7). We were able to identify four other products besides the "Tivoli Asset Discovery for z/OS" from IBM, currently used at Volvo IT Mainframe department. The products include: P-Tracker, BMC Atrium Discovery, XBridge DataSniff, and Tibco Mainframe Service Tracker. All of the reviewed SAM tools

provide the required functionality, which allow the support of the ISO/IEC 19770 (2012) guidelines for ensuring SAM.

Regarding the comments from interviewees, most of them were satisfied with the tool from IBM. Despite some issues with report accuracy, they think it performs the tasks well. However, almost all respondents expressed the need for more details in the reports, produced by the tool. Creating own software ID tags, even though technically possible, is not recommended by vendors and we would suggest to keep this feature as a topic for future work. Other requests, such as limited access to e.g. financial information for particular groups, could be implemented by creating user groups with different rights, yet not available in any of the tools. Product categorization by importance could be reached using the feature of location tagging. It is supported by all of the reviewed tools.

In general, the current SAM tool “Tivoli Asset Discovery for z/OS” from IBM tends to be a good choice for our studied organization. From the present perspective, a change of SAM tool would only be required in an instance of the organization deciding to move software assets to a cloud based setup or if the need for multi-platform support would emerge. The organization might also benefit financially from changing the vendor to find cheaper SAM tools, however investigating the specific financial information is out of the scope. Furthermore, the respondents mentioned the need to be proficient in using the tool and correctly interpret the data. Therefore, we would recommend holding additional training for the people involved, in using the SAM tool in order to make them feel comfortable with it.

VI. CONCLUSION

The aim of this study was to identify the challenges, that large-scale organizations face when managing software assets and investigate the existing SAM tools, which could help to mitigate those challenges.

To answer the research questions, we performed a case study at the Mainframe department in Volvo IT. Along with the case study, we reviewed the available literature sources on SAM in order to create a theoretical background. The study revealed that there is a lack of relevant academic research within this field, especially in regards to mainframe environments. Therefore, we followed the ISO/IEC 19770 standard for SAM.

At Volvo IT, we conducted nine interviews with people involved in different aspects of the software lifecycle management. After analyzing the results, we grouped our findings into several themes, where we identified the challenges, described the perceived benefits of SAM and presented interviewee’s suggestions for SAM tool improvement. In addition, we performed a comparison of five SAM tools.

Our study concluded with discussing the managerial and organizational challenges and benefits of SAM as identified

in the literature and interviews. The synthesis between interview results and the academic sources allowed us to visualize how people in IT industry perceive the introduction of the SAM concept, described in academic papers. Finally, we suggested Volvo IT Mainframe department to conform to the ISO/IEC 19770 standard and use a few models in order to ensure an effective SAM and improve the software lifecycle management process. We also concluded that the currently implemented SAM tool from IBM actually fulfills the basic needs. However, more time should be invested in teaching the employees how to use it efficiently.

A. Future Work

As for the future work, there are a wide range of possibilities. Knowing the lack of academic research in this field, further studies within mainframe environment at various organizations could be conducted in order to investigate situations regarding SAM and compare the results between organizations. This could lead to creating an overall model, conforming to the ISO/IEC 19770 standard, for implementing SAM in large organizations. Furthermore, an in-depth study should be conducted on the capabilities of SAM tools and critical improvements or a prototype would contribute significantly, especially within the mainframe environment. Finally, the capabilities of Software as a service (SaaS) as well as the combination of mainframe and cloud computing could be investigated if the Volvo IT Mainframe department would consider implementing cloud services.

VII. ACKNOWLEDGEMENT

Apart from the efforts from ourselves, the success of any research study depends largely on the encouragement and guidelines of many others. We take this opportunity to thank Volvo IT, Mainframe Departments: IMO Mainframe Base, IMO Mainframe z/Open and Databases, IMO Mainframe SOE Runtime Management, IMO Mainframe Transactions, Gothenburg, especially Michael Almqvist Näss, Lasse Lavebäck and Anders Bäcklund for all their support and guidance with providing ideas and feedback during the research. Special thanks to Ana Magazinius at IT University of Gothenburg for her time and assistance in the supervision process. In addition, we would like to thank Lars Pareto for his guidance and direction of this research paper. Conducting this bachelor thesis was indeed a rewarding experience.

REFERENCES

- Ben-Menachem, M., Marliss, G.S., 2005. IT Assets—Control by Importance and Exception: Supporting the “Paradigm of Change”. *IEEE Software*, 22(4), pp.94–102.
- Ben-Menachem, Mordechai, 2008. Towards management of software as assets: A literature review with additional sources. *Information and Software Technology*, 50(4), pp.241–258.

Ben-Menachem, Mordechai; Gelbard, R., 2002. Integrated IT Management Toolkit . 45(4).

Ben-Menachem, Mordechai; Marliiss, G.S., 2004. Inventorying Information Technology Systems : Supporting the "Paradigm of Change". *IEEE Software*, pp.34–43.

Booth, W. C., Colomb, G. G., & Williams, J. M., 2003. *The Craft of Research*, second ed. Chicago: University of Chicago Press.

Bott, M.F., 2000. Software as a corporate asset. *IEE Proceedings - Software*, 147(2), p.31.

Braun, V. & Clarke, V., 2006. Using thematic analysis in psychology. *Qualitative Research in Psychology*, 3(2), pp.77–101.

Creswell, J.W., 2009. *Research Design: Qualitative, Quantitative, and Mixed Methods Approaches* 3rd ed., London: Sage Publications.

Ebbers, M., Kettner, J., O'Brien, W., & Ogden, B. (2011). *Introduction to the New Mainframe: z/OS Basics*. IBM, International Technical Support Organization.

Holsing, N.F. & Yen, D.C., 1999. *Software asset management : Analysis, development and implementation*.

International Standard ISO/IEC 19770-1:2012 - Information technology - Software asset management - Part 1: Processes. , 2012. Available at: http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=33908.

Jakubijpka, M., 2010. Software asset management. *26th IEEE International Conference on Software Maintenance*, pp.1–2.

Kitchenham B, 2007,"Guidelines for performing Systematic Literature Reviews in Software Engineering", Version 2.3 EBSE Technical Report EBSE-2007-01 Software Engineering Group School of Computer Science and Mathematics Keele University Keele, UK.

Klint, P. & Verhoef, C., 2002. Enabling the creation of knowledge about software assets. *Data & Knowledge Engineering*, 41(2-3), pp.141–158.

McCarthy, M. a. & Herger, L.M., 2011. Managing Software Assets in a Global Enterprise. *2011 IEEE International Conference on Services Computing*, pp.560–567.

Microsoft, 2006. *Software Asset Management Inventory Tools: Essential to Software Asset Management Program*.

Robson, C. (2002). *Real world research: A resource for social scientists and practitioner-researchers* (Vol. 2). Oxford: Blackwell.

Sharifi, M. et al., 2009. A Novel ITSM-Based Implementation Method to Maintain Software Assets in Order to Sustain Organizational Activities. *2009 Third UKSim European Symposium on Computer Modeling and Simulation*, pp.274–280.

1. What are your current responsibilities?
2. How does your company currently administer the purchase and maintenance of its software assets (lifecycle process)?
3. What are the main challenges you perceive in managing your software assets?
4. How would you benefit by implementing SAM strategy in your organization?
5. What challenges would you possibly face if you tried to implement SAM?
6. What is your opinion about using commercial SAM tools to manage your software resources? Do you use or would you consider using any of them?
7. What do you expect from a good SAM tool?
8. Anything else you would like to add?

APPENDIX A

This appendix provides the interview guide, which was used to conduct all interviews in this study: