ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
UNIVERSITÉ DU QUÉBEC

THESIS PRESENTED TO
L'ÉCOLE DE TECHNOLOGIE SUPÉRIEURE

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR
THE DEGREE OF DOCTOR OF PHILOSOPHY
Ph.D.

BY
DAVID DÉRY

SOFTWARE ASSET MANAGEMENT PROCESSES AND MODEL

MONTREAL, JANUARY 10TH, 2008

THIS THESIS HAS BEEN EVALUATED
BY THE FOLLOWING BOARD OF EXAMINERS

M. Alain Abran, Thesis Supervisor
Génie logiciel et des technologies de l'information à l'École de technologie supérieure

M. Witold Suryn, President of the Board of Examiners
Génie logiciel et des technologies de l'information à l'École de technologie supérieure

M. François Coallier, member of the board of examiners
Génie logiciel et des technologies de l'information à l'École de technologie supérieure

M. Ettore Merlo, member of the board of examiners
Département de Génie Électrique (DGEGI) de l'École Polytechnique de Montréal

THIS THESIS WAS PRESENTED AND DEFENDED
BEFORE A BOARD OF EXAMINERS AND PUBLIC
OCTOBER 30TH, 2007
AT ÉCOLE DE TECHNOLOGIE SUPÉRIEURE

PROCESSUS ET MODÈLE DE GESTION DE BIENS DE LOGICIEL

David Déry

SOMMAIRE

L'industrie doit maintenant porter attention à la gestion de ses biens de logiciel et à leurs licences: en effet, au fil des ans, les organisations ont acheté une quantité importante de logiciels et elles doivent maintenant gérer les coûts qui y sont associés tout en s'assurant que les termes et conditions des licences soient respectés.

Jusqu'à maintenant, l'industrie avait offert des solutions partielles à la gestion des biens de logiciel, et ce en utilisant des approches différentes, des terminologies différentes et des outils avec une couverture disparate de fonctions. L'industrie s'accorde sur le besoin d'améliorer la gestion des biens de logiciels mais ne s'entend pas sur la façon de le faire. Cette thèse propose une définition de ce qu'est la gestion des biens de logiciels, fournit une analyse descriptive et une méthode d'évaluation de l'organisation face à ces processus afin que l'organisation puisse se servir immédiatement des processus. Afin de s'assurer, dans le cadre de ce projet de recherche, d'un alignement avec les besoins de l'industrie et afin d'avoir accès à un panel d'experts, l'auteur de cette thèse a participé activement, en parallèle à ses travaux de recherche, à la rédaction de la norme ISO/IEC 19770-1 sur la gestion des biens de logiciels.

Les objectifs de cette recherche sont de:

1. Contribuer activement au développement et au contenu d'une norme internationale ISO sur la gestion des biens logiciels (ISO/IEC 19770-1).
2. Capturer, identifier et analyser les éléments pertinents pour la gestion des biens de logiciels, incluant tous les éléments qui n'ont pas été inclus dans la version finale de la norme internationale.
3. Fournir une analyse de la norme internationale sur la gestion des biens de logiciels en incluant dans l'analyse les 27 processus définis dans ISO/IEC 19770-1.

4. Développer une méthode exploratoire d'évaluation permettant aux organisations d'identifier leurs écarts face au standard ISO/IEC 19770-1.

L'approche choisie a été d'aligner le travail de recherche avec la démarche en démarrage d'un groupe ISO mis sur pied en 2002 pour pallier à un certain nombre de ces lacunes et de contribuer activement au développement d'une norme sur la gestion des biens de logiciel, soit le développement de la norme ISO/IEC 19770-1.

Les résultats décrits dans cette thèse sont les suivants :

1. La construction d'un ensemble de processus pour définir la portée et le contenu de la gestion de ses biens de logiciels. Ceci permet à l'industrie d'avoir un point commun de référence en termes de contenu et de vocabulaire sur la gestion des biens de logiciels.
2. Cette thèse a permis de constater que les manufacturiers de logiciels ne s'entendent pas sur la portée ou même le vocabulaire utilisé pour décrire la gestion des biens de logiciels. De même, le rôle du gestionnaire des biens de logiciels ne fait pas l'unanimité dans l'industrie. Cette thèse adresse ces deux points via la définition des processus reliés à la gestion des biens de logiciels .
3. La thèse analyse la norme ISO 19770-1 sur la gestion des biens de logiciel afin de fournir une description approfondie de la norme face à l'infrastructure informatique et face aux autres processus déjà existants tel qu'ISO/IEC 20000 sur la gestion des services. Cette analyse est nécessaire à l'interprétation des résultats d'une évaluation.
4. La thèse propose également une façon pour les organisations de s'évaluer en utilisant des niveaux de maturité des processus de la norme ISO/IEC 19770-1; pour cela une autre norme est utilisée, soit la norme ISO/IEC 15504, pour la construction des niveaux d'évaluation.
5. Les organisations reconnaissent qu'une mauvaise gestion des biens en logiciel représente un risque pour l'organisation. Cependant, les organisations n'avaient pas de référence commune pour évaluer ce risque. L'application de l'évaluation d'une organisation en utilisant la norme ISO/IEC 19770-1 permet d'identifier la maturité des points de contrôles et de mieux identifier son impact sur l'organisation.

SOFTWARE ASSET MANAGEMENT PROCESSES AND MODEL

David Déry

ABSTRACT

The industry must now focus on software assets in order to improve the management of purchased software and their associated licenses: over the years, organizations have indeed purchased a significant amount of commercial software and they now have to manage their related costs while ensuring that the license's terms and conditions are respected.

Until now, the industry has been offering incomplete solutions to the management of software assets while using different approaches, terminologies and tools with varying functional scopes. The industry recognizes the need to improve Software Asset Management (SAM) but does not agree on the means to do so. This thesis proposes to start with a common industry SAM definition. To help organizations use the processes that constitute the SAM definition, a descriptive analysis of the processes, an assessment method and a graphical representation are provided to facilitate its use in the industry. Furthermore, to ensure the set of processes reflect the view and needs of the industry; the author actively participated in the writing of the ISO standard on SAM: the panel of experts contributing to ISO also provided a mean to validate several of the SAM topics discussed in this thesis.

The research objectives are to:

1. Actively contribute to the development and to the content of the ISO international standard on SAM (ISO/IEC 19770-1).
2. Capture, identify and analyze elements that are relevant to SAM, including those that would not make it into the final version of the international standard.
3. Provide an analysis of the international SAM standard with respect to the 27 processes within ISO/IEC 19770-1.

4. Develop an exploratory assessment method to allow organizations to determine their gaps against ISO/IEC 19770-1

The approach selected was to align the research work of this thesis with the then new ISO working group created in 2002 to address issues related to the management of software assets and to contribute actively to the development of an international standard on SAM processes, that is: ISO/IEC 19770-1.

The results of this thesis are:

1. A common set of processes to describe the scope and content of SAM. This allows the industry to have a common point of reference and vocabulary when referring to SAM.
2. Through a literature review covering both the industry and the research community it was possible to highlights the divergence of scope and terminology with software manufacturer and the lack of agreement of what is a SAM manager. This thesis addresses these issues by identifying the full set of SAM processes.
3. The thesis analyses the standard used as the basis of reference for the assessment, that is: the ISO/IEC 19770-1 standard on SAM. The description and analysis of this standard allows for a better understanding of the purpose of each process and the interactions across existing standards such as ISO/IEC 20000 on Service Management.
4. The thesis also proposes a method to assess and assign a maturity level to each of the processes of the ISO/IEC 19770-1 standard; the ISO/IEC 15504 standard is used to perform the assessment.
5. Organizations recognize that poor management of software assets puts the organization at risk. However, organizations did not have any common way of assessing these risks. With the use of the ISO/IEC 19770-1 standard and the assessment method, organization can now identify the maturity levels of control points and assess their impact on the organization.

ACKNOWLEDGEMENTS

I would like to thank Professor Alain Abran for accepting to supervise my thesis, for introducing me to the world of ISO standards and for his ongoing support and follow-up even though my day to day job has brought numerous delays and constraints throughout the writing of this thesis.

I would also like to extend my thanks to my parents, Roger Déry and Anita Trépanier for their ongoing support during this long adventure!

TABLE OF CONTENT

LIST OF TABLES

LIST OF FIGURES

LIST OF ABREVIATIONS AND SYMBOLS

ACM	Association for Computing Machinery
BS	British Standard (i.e. BS 15000)
BSA	Business Software Alliance
CI	Configuration Item
CobiT	Control Objectives for Information and related Technology
CMDB	Configuration Management Data Base
CITAMUG	Canadian Information Technology Asset Management User's Group
DSL	Definitive Software Library
FAST	Federation Against Software Theft
IBSMA	International Business Software Manager Association
IEEE	Institute of Electrical and Electronics Engineers
IDC	Consulting and Advisory Services ; www.idc.com
IEC	International Electrotechnical Commission
IMAC	Installation, Move, Add, and Change
ISO	International Organization for Standardization
ITAM	Information Technology Asset Management
IS/IT	Information System / Information Technology
ITIL	Information Technology Infrastructure Library
itSMF	IT Service Management Framework
ECP	ECPweb.com (ECP), an independent publisher and conference producer
OGC	Office of Government Commerce
MOF	Microsoft Operations Framework
SA	Software Assets
SAM	Software Asset Management
SC7	Sub Committee 7

SCC	Standards Council of Canada
SIIA	Software & Information Industry Association
SIS	Swedish Standard Institute
SWEBOK	Software Engineering Body of Knowledge
UKAS	United Kingdom Accreditation Service
Y2K	Year 2000

INTRODUCTION

Motivation

At the beginning of this PhD research work in 2003, the information systems and information technology (IS/IT) industry was still recovering from the year 2000 (Y2K) problem and the ".com" bubble. Budgets allocated to IS/IT were tight and management demanded much better control over IS/IT costs. Manufacturers of hardware and network components often provided tools and processes to manage and control their products. However, when it came to software, software manufacturers did not often provide defined and automated means of managing commercial software entitlements, that is, the rights and constraints of using software developed by an external software vendor. This meant that much of the effort to manage and control commercial software entitlements was manually intensive especially when these entitlements could be significantly different from one software vendor to another. This also suggested that the management of software assets was much less mature than the management of hardware and network components where manufacturer were offering automated solutions for assets management.

This lack of management control over software entitlements assets was becoming an increasing concern for senior management, more so that an industry trend in the 1990s had pushed organizations to buy commercial software products instead of building them in house. Purchasing off-the-shelves software was then considered "best practice", that is, not reinventing the wheel by using existing commercial software and modules instead of programming and building "in-house" solutions. This also meant that the proportion of the IS/IT budget dedicated to the purchase and maintenance of these commercial software kept increasing year after year.

This combination of increasing costs related to the management of software assets and the apparent lack of defined and/or automated solutions on how to confirm entitlements to vendor of licenses had created specialty consulting services and third party tool vendors, each proposing their own terminology, scope of issues tackled and proprietary solutions.

Such diversity of solutions and diversity of coverage of issues tackled by different vendors is then more confusing than helpful to the average software asset manager. Within the context of this diversity of solutions, this research work aims to improve the management of software assets by better defining it, modeling it and developing management models.

Research motivation and goal

The research motivation for this thesis is first to understand why Software Asset Management (SAM) was not well defined within the software industry and next to contribute to the improvement of software asset management by integrating, in particular, existing assets management models and techniques from related knowledge domains, including of course software engineering.

To pursue this research goal several steps are required initially to better understand and tackle the problem of SAM. In particular, it is important to understand how SAM is defined in several contexts such as:

- SAM for the industry (software practitioners);
- SAM for software engineering research;
- SAM for other engineering research fields.

The following literature review provides an overview of what exists in terms of industry and research publications on the topic of the inventory of assets in software engineering as well as in other related fields.

CHAPTER 1

RELATED WORK

1.1 Context

Before suggesting improvements to the management of software assets it is important to know and be aware of what has already been published and proposed both by the industry and by the research community. This literature review describes and assesses the current situation and level of knowledge on asset management in software engineering as well as in other engineering fields from which additional knowledge could be leveraged.

1.2 SAM in industry

1.2.1 Sources of information in the IS/IT industry

1.2.1.1 Industry white papers

The IS/IT industry is well known for its abundance of industry white papers and reports from independent consulting organizations. Some of the most recognized sources of such white papers and reports in the IT field are:

- Gartner Group: research notes (2001-2003) [1-10];
- Meta group (2000-2001): opinions and observations (now part of the Gartner Group). [11]
- Giga [12, 13]

These consulting organizations have published a number of reports and white papers on inventory management, software discovery and asset management; this is an indication that the industry is interested in this SAM issue. Sometimes these consulting organizations also provide survey data and projections on where the market is heading to. These consulting organizations also often discuss the status of the industry, how to use ITIL (Information Technology Infrastructure Technology) [14, 15] and where ITIL falls short.

However, the expression "Asset Management" is not used the same way in most of these industry reports and its meaning and scope appear to vary from a consulting organization to another. In addition, these industry-produced white papers usually do not document how such survey data are obtained nor what share of the industry has been sampled: these industry reports are therefore not verifiable and not reproducible both in terms of process and in terms of results obtained. The usefulness of these reports for research analysis purposes is then very limited (It is to be noticed that most of these reports are not in the public domain – a list is presented in APPENDIX I).

Notwithstanding, these industry white papers refer to some recurring themes, as listed in Table 1. Amongst these recurring themes, most appear to agree that ITIL is a good starting point for SAM but that it is not sufficient; they do not, however, agree on how to address the shortcomings of ITIL with respect to SAM.

Table 1

Consensus and disagreements in industry reports - from APPENDIX I

The industry agrees that	The industry does not agree on
1. ITIL is a good starting point. 2. ITIL does not adequately define what is SAM. 3. Tools are not the solution; the process must be defined first. 4. Maintaining the inventory is only a subset of SAM.	1. How to complement ITIL to include SAM. 2. A common definition for SAM. 3. The scope of SAM; what is included and what is not.

1.2.1.2 Tool vendors

The IS/IT market offers a number of software solutions (see APPENDIX II) and organizations such as BSA [16] and Microsoft's SAM best practices group [17] maintain a list of SAM consultants and SAM tool vendors. Some software vendors offer software tools that maintain an asset management repository (such as Remedy, Peregrine Systems, Provance Technologies) [4], some offer inventory and discovery tools functionalities (Microsoft, Tivoli Systems, Peregrine Systems, Tangram Enterprise Solution, Tally Systems, Computer Associates) [4] while other vendors propose some configuration management functionalities (Computer Associates, Microsoft, Novell, Tivoli, Veritas) [4]. This is, of course, not an exhaustive list of vendors as they keep changing over time with the emergence of new companies and the merger and acquisition of existing ones. But while the different software solutions all use the term "SAM tool", they do not, however, perform the same set of subtasks; this further contributes to the confusion about SAM and its definition and scope.

1.2.1.3 Classifications

Some effort has been made to classify SAM tools. Some white papers and industry analysts group these tools into 3 categories [18] (see APPENDIX III for details):

- Inventory tools
- Asset Repository
- Software usage.

However, there are some differences as to what is included in each category. For example, the expression 'Inventory tools' is often a synonym for an 'auto discovery tool', but the scope and sophistication of each SAM tool vary greatly and it does not appear fair to group all of these software under the same label.

In practice, SAM tools may involve the following functionalities (this is not an exhaustive list since the industry offers several definitions and uses a number of labels to describe the same set of subtasks):

1. **Discovery**: In order for a software asset to be discovered, it must be available on the network at the moment of the discovery exercise. The discovery of the software only identifies the presence of the software (i.e. filenames); it does not tell anything about the nature of the software, including its commercial product name.
2. **Identification**: once a software has been discovered, it must be identified according to its commercial product name at the time of purchase. This can be quite complex since a software may change in nature over time with the addition of patches and fixes. The nature and the labeling of the software may also change over time: it is not clear how this monitoring is being performed since it may include a comparison algorithm to a proprietary library and may require subscription fees in order to have access to this proprietary library and identification scheme (which locks the buyer to this specific vendor solution).

3. **Software Usage**: The usage of the software must be monitored in order to determine and record how it is used. This information can be used to support and validate the matching process. There is, however, no clear definition of what constitute software usage as it may vary from vendor to vendor: a window that is opened does not necessarily mean that the software is used, and the CPU usage by a software is also not accurate.
4. **Entitlement** (e.g. license terms and conditions are respected): commercial software are licensed to the buyer and this limits the usage of the software according to specific terms and conditions; these terms and conditions may change overtime in order to maintain or increase profits for the software license owner. However, the terms and conditions can be difficult to monitor since they can be based on factors that are not easily measurable. Software vendors often provide no automated means to measure, nor detailed instructions on how to verify compliance to these terms and conditions. These terms and conditions can include the number of concurrent users, the usage time, the number of named users, the number of CPU used or any combination of these factors. All of these terms and conditions can change overtime and from platform to platform for the same software. Software vendors do not provide an automated way to do validate compliance and organizations that use these licensed software are not ready to let the software vendors have complete access to their infrastructure to verify compliance.
5. **Reporting**: a simple report on usage or entitlement may not be enough if both parties cannot agree on a common way to measure software usage or entitlement compliance against terms and conditions. This means that the type of information to collect and the conditions under which this information is collected need to be agreed upon with the owner of the software license; this is rarely done. Although the reasons are not well documented, it appears that the lack of clear instructions from the software vendors on what

constitutes an irrefutable proof of compliance only complicates the process of compliance against the license's terms and conditions.

Currently, tool vendors refer to Asset Management when using any combination of these functionalities: this contributes to the confusion surrounding Asset Management.

Even though software vendors attempt to differentiate their products from one another, they do agree on some points: alignment to ITIL is a marketing advantage - however, usage of the term "ITIL compliant" does not appear to have a universal meaning. **Table 2** lists some of the most common themes discussed by SAM tool vendors and are presented in two columns: the items on which tool vendors agree upon (left column) and those items that do not make consensus (right column); ITIL is at the center of much of the discussions.

Table 2

Tool vendors: Agreements and disagreements

Tool vendor agrees on	Tool vendors disagree on
1. Alignment with ITIL is an important marketing advantage.	1. The meaning of ITIL compliant.
2. The importance of using a Configuration Management Database (CMDB); the list of configuration information items and its relationship to other components (ITIL) is often required by the compliance process.	2. The number and types of software functionalities to include in inventory management, software discovery and other commonly used "functionality" Labels.
3. There is no consensus or constraint on where to store and show financial and asset management (i.e. contract entitlements) information.	3. The labeling of data and grouping of data related to Software Assets (i.e. financial

Tool vendor agrees on	Tool vendors disagree on
	data, contract information, inventory information and terms and conditions of the license),

1.2.1.4 SAM books

There are few books on the topic of SAM. For instance, a search on Amazon.com shows a list of industry white papers (see section 1.2.1.1), books on how to use a specific SAM tool (see section 1.2.1.2) and a book by ITIL on SAM. [19]. The ITIL book on SAM defines SAM as "all of the infrastructure and processes necessary for the effective management, control and protection of the software assets within an organization, throughout all stages of their lifecycle". This book published by ITIL promotes SAM as good corporate governance: the organization's roles and responsibilities puts emphasis on the creation of a SAM database that would be part of the CMDB of ITIL. Amongst its core processes, such processes as software identification, asset control and status control [19] are presented to identify, control changes and report on changes.

However, a single industry book (e.g. from ITIL) on this topic is not enough to conclude that there is a consensus in the industry.

1.2.2 Industry references

1.2.2.1 References

Industry references, and standards in general, play an important role in engineering disciplines since they act as official points of reference and ensure that the profession has a common understanding of what is expected of the profession. Such industry references and standards imply, for instance, that the engineering profession is mature enough to have a recognized authority set up to establish those standards.

In the software engineering discipline, the governing bodies capable of creating and overseeing standards are mainly represented by IEEE and ISO (see APPENDIX V: list of Software engineering standards by IEEE and APPENDIX IV: list of Software engineering standards by ISO)

Although several standards target software issues, none specifically covers the management of software assets. Some covers the acquisition of software in general (such as IEEE Std 1062-1993) and its maintenance (IEEE Std 1219-1993) but none discusses how to manage licenses, monitor usage and how to reconcile inventories with license entitlements.

A de facto industry standard has also emerged with ITIL. This industry reference is divided into two subsets of processes: processes for Service Support (e.g. Maintenance) and processes for Service Development. Although this industry reference is widely known in the industry, this document is rather vague on how to manage software assets: the term 'Assets' is sometimes mentioned but it is not defined: APPENDIX VI lists the occurrences of the term "Asset" and the term "license" within the ITIL Service Support and Service Development books; the lack of details on these terms has therefore led the consulting industry to provide several alternate definitions (see APPENDIX I and APPENDIX II).

In summary: the industry does not agree on a common solution for the management of software assets and the diversity of proprietary solutions proposed by consulting organizations and the lack of standards for SAM are indications that the industry was not sufficiently mature in the early 2000 to have standards on what is SAM and how to perform it.

1.3 Asset Management in Software Engineering

1.3.1 Context

The IS/IT industry has evolved and so have its research interests. At the beginning of the computing industry, hardware was very expensive and research was investigating how to use the hardware in the most cost efficient way possible. As the cost of hardware came done, the importance of other research topics grew; one of these was the construction of software and, more recently, the management of software assets. The following sections depict this evolution by taking a historical perspective on inventory management.

1.3.2 Historical perspective

The Information Technology (IT) service industry, although relatively young, has changed considerably since its origins in the 1950s. The initial focus of IT services was on hardware only and it progressively moved towards software and, later, into services as indicated in figure 1 from [20]. In the 1990s, clients of IT services started to require a complete set of services and various approaches were developed to attempt to meet these needs, such as SAM (Software Asset Management), ITAM (Information Technology Asset Management) and end-to-end solutions.

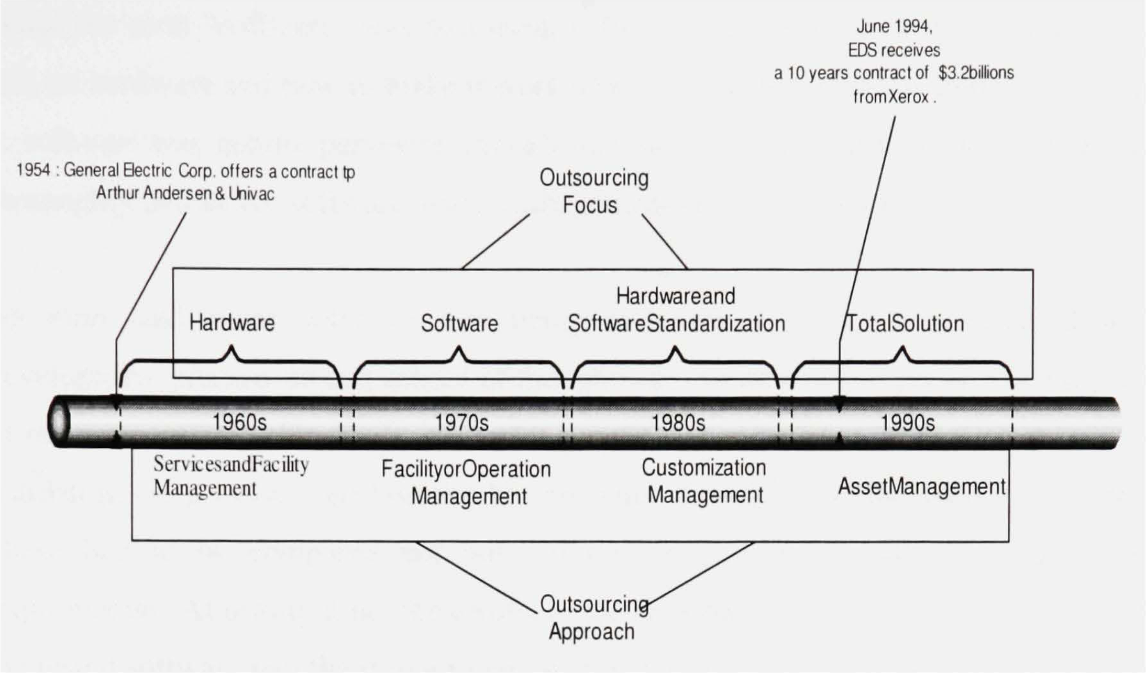


Figure 1 IT Industry outsourcing/services history adapted from [20]

In parallel, the research community developed new knowledge which was being referred to as software engineering, but which had not yet gotten into the mainstream of engineering communities. In the mid 1990s the IEEE-Computer Society initiated a project, referred to as SWEBOK [21] (Software Engineering Body of Knowledge), to define the software engineering discipline by developing an international consensus on its related areas of knowledge. The SWEBOK now includes several knowledge areas that are directly linked to the lifecycle of software (Software requirements, Software design, Software Construction, Software Testing, Software Maintenance) as well as other knowledge areas that are less specific to the construction phases of software; these knowledge areas can interact more directly with software acquisition and other parts of the organization such as Operations Management (Software Configuration Management, Software Engineering Management, Software Engineering process, Software Engineering tools and methods, Software Quality).

When the term "software" was first used in 1952 [21], the focus of the IT industry was still on hardware and how to make it work properly. By the 1970s, as indicated in Figure 1, software was getting pervasive enough that the need arose to manage software more thoroughly and hence software assets management related contracts emerged.

As more and more software was being built, more rigour was required in the development process. In one school of thoughts, the emphasis was put on the traceability of requirements: a software development project was composed of several phases, each validating the previous one (similar but not limited to the waterfall model). The design phase had to be compared not only to the analysis phase but also to the initial requirements. At testing time, the results of the tests had to demonstrate to the client that the tested software met the initial requirements. Several tools were developed to manage code and to manage requirements. The top of the line tools could be used throughout every phase of the development process and would ensure that the code and the implemented solution satisfied the initial requirements: these tools were to ensure traceability from requirements to implementation.

However, the context described above changes considerably when the software is already developed (i.e. a commercial product) and is used by a buyer under license from the license owner: the license purchaser (i.e. the organization using the commercial product) needs to manage, control and monitor the usage of this software that they do not own. This requires processes and tasks that are very different from ones used in the development (e.g. construction) phases. This means that the license user of the commercial software will need to manage license fees and ensure the license owner that license compliance is being met throughout the year (or contract length).

The license user will also need to monitor and follow the assignments of software from one resource to another one; here, a resource can be individuals or hardware. Any movement of people or hardware (commonly known in the industry as IMAC:

Installation, Move, Add, Change) will need to be tracked and any software ownership change will need to be updated in the appropriate repository. If the configuration of a particular hardware is changed, the software purchaser will need to verify with the software vendor if a new software license agreement is required. If a license agreement is breached, the software purchaser that is under license must negotiate with the software vendor or pay a fine. All of this has little to do with the traceability of the requirements of the software`s functionality as it is the case in software development.

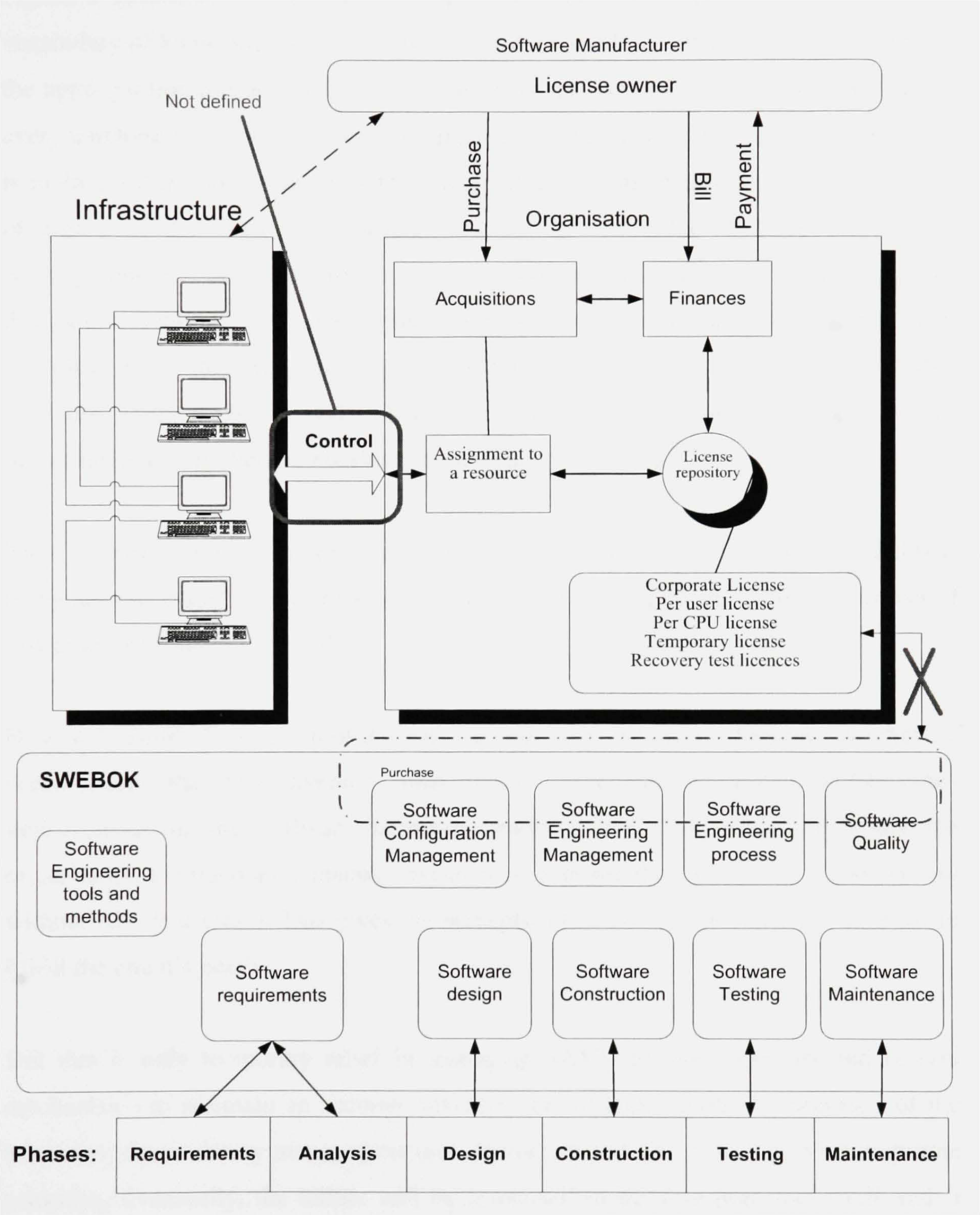


Figure 2 Licensing and software engineering

Figure 2 illustrates the dichotomy between the software engineering's (SWEBOK) vocabulary of knowledge areas and the terminology used by software asset managers. In the upper portion of the figure, the asset manager interacts with the external vendor and every purchase has to balance out with financial records. In addition, a license repository is maintained and kept aligned with current software assignments. In the lower portion of figure 2, the software engineering areas of knowledge describe the various development (e.g. construction) phases of the software lifecycle while other operational disciplines, delimited by a dotted line, are more development phase independent. It is not clear from this diagram, from SWEBOK and from the software engineering literature (referenced in SWEBOK) where does software asset management fits, including inventory, license entitlement and usage monitoring.

The IT services industry's need for Software Asset Management is growing and this is being acknowledged by the software industry that is developing an important number of inventory management tools[4].

However, simply buying a tool does not provide any sustainable solution. In practice, organizations that buy inventory management tools initially observe a temporary improvement in the software inventory tracking: the purchase itself forces the organization to undertake a manual inventory to populate the tool (i.e. an inventory tool without data is useless). This gives the perception that the tool is sufficient by itself to fulfill the client's needs.

But this is only temporary relief in managing SAM: without processes and control mechanisms to maintain an accurate inventory of software assets, the accuracy of the content of the inventory management tool slowly fades as people shift to other corporate priorities. Eventually, the failure will be associated to the inventory tool itself, rather than the lack of process, and a new tool will be bought. By buying a new tool, the entire inventory exercise will start all over again: a simple inventory is thus not sufficient for SAM.

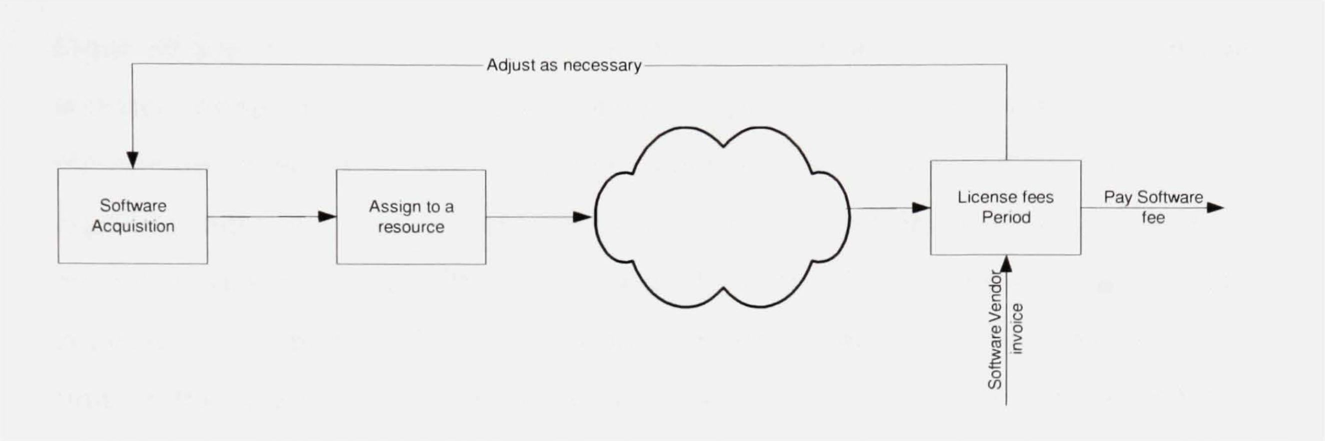


Figure 3 Management without adequate monitoring

Figure 3 illustrates the source of many disappointments from a financial point of view: if a software is not properly tracked, it will be lost from management oversights and will have to be 'rediscovered' when the software vendor sends a bill. Since some software license fees can be quite large, such lack of monitoring and controls can be expensive. As indicated previously simply purchasing a new inventory tool is not sufficient

In most popular software engineering textbooks, the focus has been on the construction of software [21-25] rather than on the management of software as assets. But market changes in the industry are increasing pressure for better software asset management procedures: these market changes are presented historically in the next section.

1.3.2.1 Managing Assets

In the management of hardware, the industry has already worked towards developing some standards: the DMTF (Desktop Management Task Force) [26] is such an example which leads to believe that the process of the hardware side of asset management can be more easily defined; automation is then easier to obtain.

From section 1.2, it was observed that there is interest in appropriately managing software assets but there is little consensus on a common solution. Furthermore, outsourcing contracts are becoming more common and outsourcing contracts lead to legal commitments: vague or subjective clauses are thus not desirable. This interest can be noted back as far as 1954 when General Electric Corp. contracted with Arthur Andersen and Univac [27] for one of the most important outsourcing contracts of its time; software assets were part of the agreement although not necessarily labelled as such then.

The term "asset management" only gained wide spread usage in the 1990s [20]. Furthermore, in some of these outsourcing contracts, there are clauses which force changes to the IT infrastructure and hence complicate further the management of software assets. One such clause is the "technological refresh" clause which ensures that the outsourcer will keep his client software related infrastructure up to date in terms of technology even though doing so increases the outsourcer's costs [28]. This means that the infrastructure must evolve and change and that the licenses and its terms and conditions may change regardless of whether it is monitored adequately or not (i.e. if the license depends on the hardware configuration, a change to the hardware configuration may require a software upgrade).

As in any decision to apply changes to the organization's IT assets, there is a tradeoff. This tradeoff for software is mostly between two choices: the number of new features versus the stability of the application. In [29], an economic model is presented that captures the various tradeoffs in software release decisions and proposes a methodology to determine the best release time for a new software: this model assumes that the value of the new functionality for the enterprise can be appropriately evaluated by the organization against the cost of the changes to maintaining it. This assumption may not always be true since the IS/IT industry is struggling with software asset costs.

The management of software assets also requires some sort of control mechanisms to restrict to whom and how software are being accessed. In the internet domain, some researchers are now focusing on digital rights management because of the pressing concerns from the online music business [30] and to ensure proper billing to people who use a specific software [31]; it is to be noticed that [30] and [31] are more concerned with techniques that would impose restrictions than with the processes needed to enforce these restrictions.

Licensing policies and models are also used to bind and inform end-users of their legal obligations. These license management models are either based on technology or based on methods and, in [32], the authors offer a model to manage not only at the software level but also at the component level (i.e. management at a lower granularity). The introduction of various licensing policies can also help organizations to maximize their revenues by exercising price discrimination through the use of different pricing schemes: for instance, one pricing schema for those buying the software and another one for those renting the software[33].

Techniques are also deployed to authenticate the owners of the software application. Theses techniques vary from the use of forward-secure signatures (FSSs) [34] to software watermarking which can have distinctive names depending on their usage: Validation Mark, Licensing Mark, Authorship Mark and Fingerprinting Mark [35]

This focus on techniques denotes an interest in finding a tool solution to the problem rather than in defining processes.

1.3.2.2 Software Engineering research

The industry has come a long way since the early days of programming when programmers would work on programs until completion, without any monitoring, supervision or formal requirements. In these early times, proposing an analysis and a design phase before coding was then already considered as a significant step to improve the software development process [36].

But in year 2000+, proposing new phases and new development lifecycle models is not enough: some practitioners find that focusing on the construction of the software itself does not necessarily bring enough value. In [37] the authors noted that much of the effort around the construction of software is focused on controlling costs but not on creating added value for the organization. This is why, a roadmap is proposed in [37] to develop fundamental knowledge that would lead to measurable objectives and create added value for the software being delivered: by demonstrating that the software is meeting specific objectives, it would be easier to demonstrate that the software provides value to the organization.

Several authors have observed an evolution in the development and use of software [24] (i.e. in terms of techniques and tools to construct software and in the growing range of software applications). Methodologies such as agile development have been introduced to provide more rapid development through strong participation from users such as extreme programming [21]. Other authors have noticed that some disciplines such as software purchasing have been neglected in the software engineering curriculum [38]. This focus on software construction and a scarcity of attention to software purchasing and the management of assets can also be observed in SWEBOK [21].

1.3.2.3 Standards in software engineering

The IT industry has proposed several standards to help organizations work better together by using a common vocabulary and by providing guidelines to improve industry practices.

Some initiatives have attempted to identify all processes an organization should have in order to reach best practices status; one of these initiatives is ITIL (Information Technology Infrastructure Library) which has documented best practices for IT service management and is based on the collective experience of commercial and governmental practitioners worldwide. It originated in the U.K. when the OGC (Office of Government Commerce) [14, 15] observing a high turnover of consultants and not wanting to loose the expertise and knowledge, decided to start capturing this knowledge under a 'best practices' umbrella.

The overall goal of ITIL is to maintain the integrity of the IT infrastructure in the most cost efficient manner. The ITIL model is composed of several processes and one service. Each process is independent of the structure, department or organizational grouping of the organization; it relies on processes and process owners that are responsible for the implementation, maintenance and compliance. The process owner must make sure that the goal of each process is reached.

ITIL processes are classified into two groups depending on whether they pertain to the development of software (Service Delivery) or to its maintenance (Service Support):

- Service support : maintaining the IT infrastructure
    - Incident Management
    - Problem Management
    - Change Management
    - Configuration Management

  - Release Management
- Service Delivery: introducing changes to the infrastructure
  - Availability Management
  - Capacity Management
  - IT Service Continuity Management
  - Financial Management
  - Service Level Management

Furthermore, each process meets a specific goal for the organization: each of these goals must be addressed to maintain the stability of the IT infrastructure listed in APPENDIX VI. This means that the industry experience states that all of these goals must be addressed if the organization wants to maintain the IT infrastructure in a cost efficient manner. In other words, not addressing one of these concerns will require other processes to compensate for the absence of one or several of them; this will create a strain and an inefficient use of resources.

Terms such as "Asset" and "Software license" are mentioned in ITIL but are not part of any of the goals of its processes: however, if the configuration management process is sufficiently mature, assets and software licensing are assumed to be managed but no indication is provided (see APPENDIX VII for more details) on how this is to be done. The focus appears to be on ensuring that the IT infrastructure is stable and operating according to agreed levels. There is no mention of ensuring that a process must be in place to maintain a software inventory or to facilitate verification of software entitlements, although in the section on roles and responsibilities of ITIL, it is mentioned that management is legally responsible for ensuring licensing conditions (see APPENDIX VII).

Although there are a number of publications on ITIL, when combining searches of ITIL with "software assets" or "software licenses" on **Compendex & Inspec** no article shows up: discussions on ITIL and its use for managing Software Assets does not appear to have reach the research community yet. There is however, some acknowledgements that ITIL processes can have several levels of maturity by applying the assessment model of ISO 15504 with ITIL [39]. This will be further developed in CHAPTER 6 when the application of SAM to the industry will be discussed.

Other organizations such as ISO and IEEE also cover the management of software, although not from a SAM perspective. For example ISO/IEC 14764 [40] on maintenance is concerned about keeping the software in a "working state" but does not cover software licenses, entitlements or proof of purchase. On the other hand, IEEE is more specific about the management of software but limits itself to the purchasing of software. For example, IEEE 1062 [41] does not cover what happens to software once it is introduced in the IT infrastructure; feedback collected are only for future purchasing references.

1.3.3 Industrial Engineering

At the beginning of the industrial revolution, employees would line up in front of the manufacturing enterprise and the first employees to show up would get the job for the day. When time and effort were spent on studying tasks and activities, it was discovered that by using the appropriate activity sequences to perform a specific task and by training employees, a productivity gain could be obtained and better product quality could be achieved. By being more formal about activities to be performed, planning and control mechanisms could be used to manage the manufacturing plants.

Figure 4 presents an overview of the evolution of industrial engineering with a focus on the fields that are more of interest to Software Asset Management, that is, those that have asset management perspectives that could be applicable to SAM.

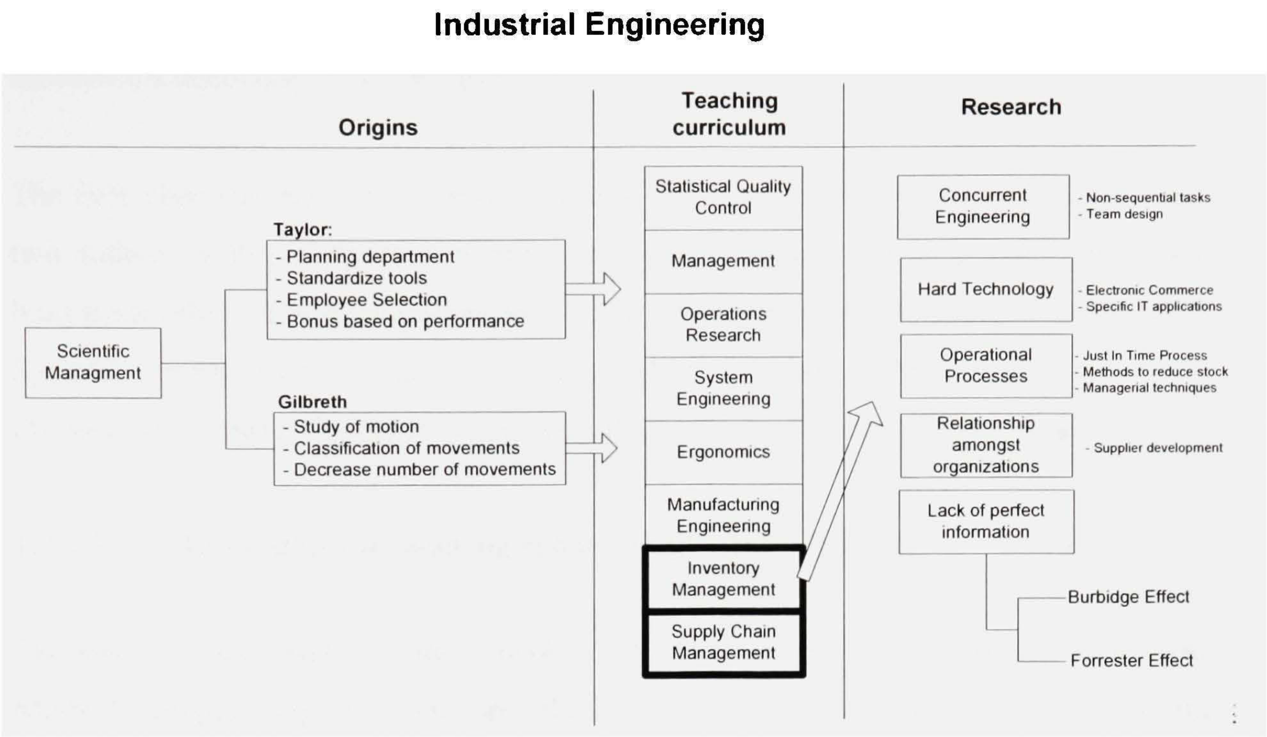


Figure 4 Evolution of Industrial Engineering - from [42]

1.3.3.1 Origins

Industrial engineering is not the oldest of the engineering disciplines: it got started only with the industrial revolution [43].

In the beginnings of the industrial revolution, authors such as Frederick W. Taylor [43] introduced the concept of scientific management which included having a planning department, standardized tools, employee selection and providing bonuses based on tasks performed in the specified time.

For his part, Frank Gilbreth [43] was more interested in the analysis of fundamental motions of human activity and classified the basic motions into 'therbligs' such as search, find, transport, empty. This is very similar to workflow terminology found in software engineering. Gilbreth studied standard methods to decrease the number of movements necessary to accomplish a task.

The first classical industrial engineering texts were extensions of the work from these two authors, with the addition of statistical quality control. Industrial engineering was built upon other ideas and methods from related disciplines such as Operations research, System Engineering, Statistics, Management Sciences, Methods Engineering, Production Planning, Ergonomics and Manufacturing Engineering.

1.3.3.2 Industrial Engineering related fields and subfields

The study of tasks and how they should be performed has evolved through time; with Methods Engineering, not only are the tasks being studied and measured but the sequences of tasks are also being optimized. In addition, Concurrent Engineering studies how best to use concurrent tasks (i.e. use of tasks in parallel). Through the use of models and simulations, flaws are removed even before they are implemented: "A process, such as productibility engineering, that disposes of flawed design concepts during team deliberations, before they become part of a hardened design, simply saves everyone involved the necessity of extricating the design weaknesses after they have become a formal part of the design." [43]

1.3.3.3 Production System Control

Production planning is concerned with determining what resources must be available on a specific site at a specific place in time to ensure that manufacturing goals are accomplished. Once the planning is done, some controls must take place to ensure that the appropriate materials reside at the designated sites to ensure that the manufacturing processes are made available in a cost-effective manner where and when needed [43].

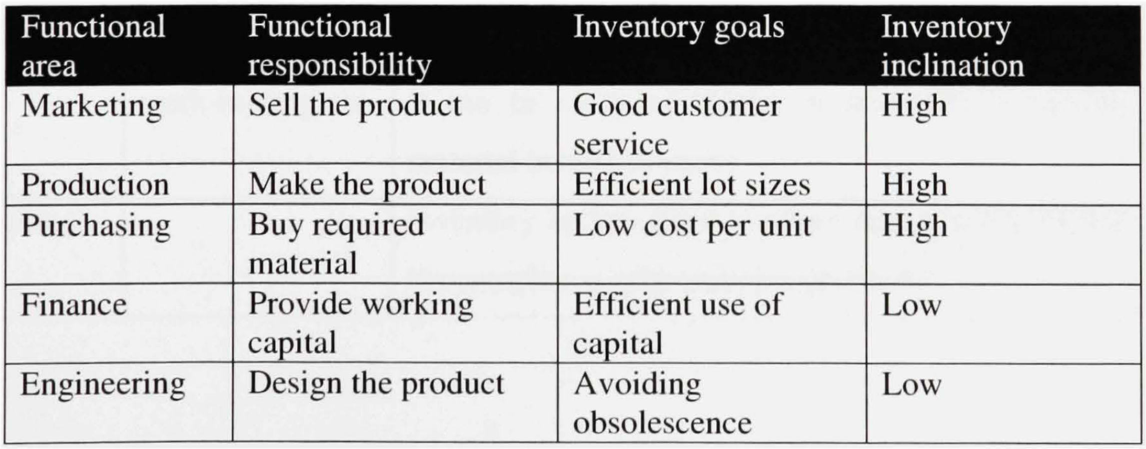
1.3.3.4 Inventory management

There was a lot of interest in inventory control in the 1970s because of the high interest rates and related costs incurred with important lot sizes, making these very expensive [43]. At that time, several models were developed to determine the optimal lot size of the optimal order frequency rate: there is a trade-off between ordering large quantities to obtain volume discounts vs. the cost of storage of unused products; several models have been proposed to address this problem which tries to minimize overall costs by selecting the optimal set of order frequency and lot size.

But the management of inventory is also impacted by the functional responsibilities; someone from marketing will try to please the customer to obtain a sale while someone from purchasing will try to obtain the best per unit price. Furthermore, some organizational functions will not focus on inventory at all: for example, finance will be more focused on the use of capital and not directly on the management of the inventory itself. **Table 3** lists some common functional responsibilities together with the related goals and inclination towards the management of inventory.

Table 3

Department orientations towards inventory [42]



Functional area	Functional responsibility	Inventory goals	Inventory inclination
Marketing	Sell the product	Good customer service	High
Production	Make the product	Efficient lot sizes	High
Purchasing	Buy required material	Low cost per unit	High
Finance	Provide working capital	Efficient use of capital	Low
Engineering	Design the product	Avoiding obsolescence	Low

Sometimes it is necessary to keep some stock because of how the manufacturing process works: if there is not enough stock the whole process could come to a stop, then some stock is needed for safety reasons. Other times, stocks are obtained because large quantities were bough to get a volume discount. **Table 4** lists various types of stocks, classified by their functional role. The first type of stock, the "working stock" is the one most referred to in the models that determine the lot size.

Table 4

Functional classification of inventory [42]

Stock name	Role	Rationale
working stock	cycle or lot size stock	inventory acquired and held so that ordering can be done in lot size: the size to qualify for discounts and/or freight rate discounts
safety stock	buffer or fluctuation stock	inventory held in reserve to protect from uncertainties and averages out to the amount of stock needed for the replenishment cycle
anticipation stock	seasonal or stabilization stock	to cope with peak seasonal demand, erratic demands (strike or vacation shutdown); it is acquired in advance of the requirements to balance production
pipeline stock	transit stock or work-in-progress	stock to allow for time to receive material; externally it can be stock on trucks or internally it can be material being processed
decoupling stock		inventory accumulated between dependent activities for complex synchronization problems

1.3.3.4.1 Economic Order Quantity

The Harris inventory model [43] was used to determine the optimal lot size as well as the order frequency. However, it is an idealized model where only a few variables are used (i.e. material cost, inventory holding cost, order preparation costs). The Harris model (also know as the Wilson model) is a deterministic model but there have also been some probabilistic variations proposed (using means and density functions for the demand of goods). Its strength and its weakness is its simplicity; however, JIT (Just In Time) models have since become more popular.

An inventory model will calculate and determine a reorder point; a monitoring scheme will be required to be able to make a decision to continue with the current lot of stock or to reorder some more stocks. **Figure 5** represents graphically the monitoring and decision making of the reordering stock based on a predetermine reorder point. For physical stocks, this can involve making a threshold line in the bin that contains the stocks and reorder when the stock level is below the threshold level indicated by the line in the bin.

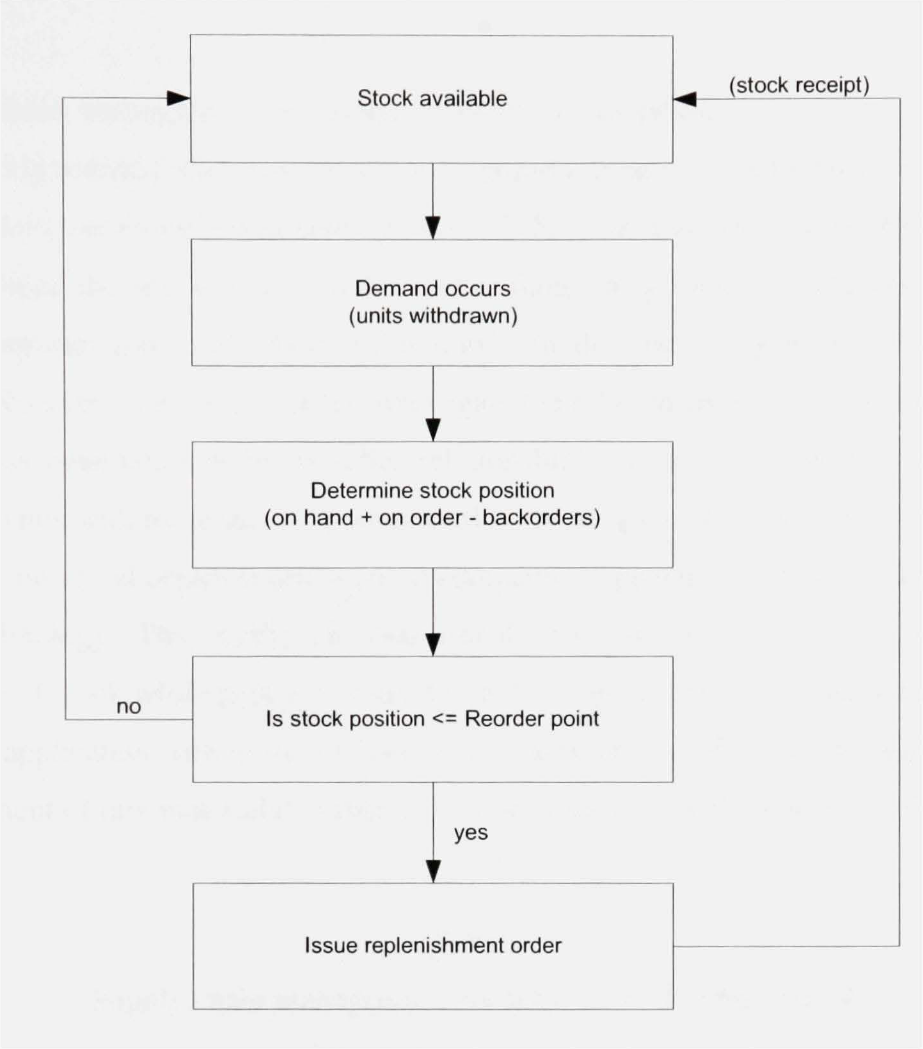

Stock available
(stock receipt)
Demand occurs
(units withdrawn)
Determine stock position
(on hand + on order - backorders)
no
Is stock position <= Reorder point
yes
Issue replenishment order

Figure 5 Reorder point determination working with fixed order stocks [42]

1.3.3.5 Supply Chain Management

Managing the production chain with its various inputs and transformation processes has led to a specific field of expertise: supply chain management. In a perfect world, the information would be available to all at the appropriate time but, in practice, this is not the case. This lack of perfect information is often represented by [44] :

- The Burbidge Effect where "noise" makes production oscillating near the target demand without truly reaching it (otherwise the information would be perfect).
- The Forrester effect which is often represented by overshooting the true demand level and hence is constantly readjusting.

Supply chain management is another industrial engineering field that has evolved significantly recently and some of the more popular trends are being discussed in "*IEEE Transactions on Engineering Management*" [45]. The authors of this IEEE magazine have grouped the research approaches under three categories; "operational processes", "social organization" and "hard technology". In the social organization category the authors observed that some authors investigate the relationships with other organizations (i.e. across boundaries or buyer/seller relationship) but few have studied the effect of supply chains with more than one supplier (i.e. the supplier of a supplier). Also, the ratio of papers on social organization is small compared to papers on operational processes or hard technology. The papers on operational processes are usually centered on the reduction of stock while papers on hard technology discuss specific electronic commerce and IT application solutions to handle or monitor specific operational tasks (i.e. management of raw materials). Table 4 presents a summarized view of the findings from [45].

Table 5

Supply chain management consensus and divergences [45]

Supply chain management category	Focus / point of interest
Operational processes	Reduction of stock
Social organization	Studied relationships across organizations but with a single supplier
Hard technology	Electronic commerce Management of raw materials (and other specific operational tasks)

Research on supply chain also includes the study and determination of when to test for quality or when to replace a part. These concerns are also found in other fields of engineering.

1.3.3.6 Rationale for the existence of inventory

According to Tersine [42], organizations are faced with inventory because the supply and demand cannot be perfectly matched. Tersine cites four factors to explain this imperfection between supply and demand, as listed in **Table 6**: these factors include time factors, the discontinuity in the production process, the uncertainty of production and economy factor obtained from volume discounts.

Table 6

Rationale for the existence of inventory [42]

Factor	Explanation
Time factor	The time required to develop the product: few people would be ready to wait for the entire process - the inventory helps reduce this waiting time.
Discontinuity factor	Allows the treatment of various dependent operations without having to force consumers to adapt to the necessities of production.
Uncertainty factor	All the unforeseen events such as errors in demand estimates, equipment breakdown, strikes, acts of gods.
Economy factor	Buying in bulk in order to reduce significantly unit cost. Inventories can be used to smooth production and stabilize manpower levels in undulating and seasonal businesses.

1.3.3.7 Inventory management limitations

As mentioned by Tersine [42], "Inventory management is everybody's concern, but it is not uncommon to find everybody's concern but nobody's responsibility: responsibility is divided among department". This is aligned with ITIL's point of view to have a process for each concern so as to make the process and the concern independent of the organization. Failing to do so would mean delegating the management of inventory to a clerical routine.

1.3.4 Engineering Research

The management of assets is a concern found in several engineering fields. Common to all those fields, including software engineering, is the observation that practitioners will tend to define techniques and automate the process regardless of their levels of maturity.

1.3.4.1 Technology specific research

Working in an industrial environment or in assembly lines, various equipments will need to be monitored and maintained. This requires a certain understanding of what to look for since the signs of wear and tear are not the same for mechanical stress, electrical stress, thermal stress and chemical stress. This is why the authors of [46], explain how wear and tear affect mechanical, electrical, thermal and chemical components and what to look for in each case. However, quantifying and predicting the component's failure is not easy because the information is largely based on anecdotal evidence; there is a poor understanding of the asset's aging process because the industry did not previously recognize the need to plan for the long term.

Instead of anecdotal evidence it would better to start with a current picture of the situation, a baseline, and determine the various conditions that a piece of equipment can take. These conditions can be monitored according to a predetermined set of parameters. This is what is proposed in [47], with a description of the possible tests that will vary from plant to plant (i.e. impedance monitoring for batteries, load readings for transformer and dielectric test for switchgears). If these are performed manually, then they can be very time consuming.

To help automate the determination of what to inspect and when to replace pieces of equipments, the authors in [48] are proposing a support tool called Risk-Based Asset Management (RiBAM). This support tool uses the Net Present Value (NPV) to select the best maintenance alternative between "overhaul", "replace" or "stop all maintenance" by using a probabilistic model (i.e. Monte Carlo simulation) and by building life curves through the 'Asset Management Planner' (AMP) to calculate the expected time to failure for the various scenarios.

The latter tool helps automate the decision making process. However, the authors in [49] are concerned with automating the reading and determination of the status of a given piece of equipment. The goal of this automation is to increase productivity through reading equipment data online thus reducing manual labor associated with the alternative. This is achieved through the addition of a new set of recorded fields, the 'Field Gateway' (FGW) which are added to the current device and form a new intelligent field device (called here the 'Foundation Field Bus' or FFBus) to transmit the information to a central repository or the 'Manufacturing Execution System' (MES).

1.3.4.2 Managerial techniques

It is not sufficient to have hardware to monitor and measure pieces of equipment; there must also be a set of managerial techniques to ensure that the information is properly used and that appropriate actions are taken. This is where authors in [50] emphasize the importance of having a good methodology and of applying it in a way that ensures alignment with stakeholders interests for cost reduction without jeopardizing the reliability and quality of the product.

This need for good managerial techniques is quite common in large organizations: for example, hospitals where the number and complexity of assets increased tremendously over time also need good asset management practices. The new Boston city hospital was no exception when, in the 1970s, the clinical equipment became more sophisticated and grew in number (over 2,700 devices) [51]. This required the establishment of procedures and monitoring techniques to determine which equipment required repair, testing, calibration, modification and installation. However, it was found that some items, that were more sophisticated, required more monitoring and more frequent corrective actions as well as a separate set of monitoring tools and processes.

Other considerations such as concerns for the environment can bring other challenges. One approach is to limit the negative effects of retiring an asset by planning to recycle as many parts as possible. This is what happened at Xerox Asset Recovery management by starting a major strategic initiative to integrate the idea of using recycled parts into the design of the Product Delivery Process (PDP) and training programs for every phase of the product's lifespan [52]. This process is also detailed in [53] and labeled Total Asset Recovery Management process.

Up until now, all asset management decisions were based on the management of a single asset at a time. However, in some industries, some assets are concurrently competing for the same resources. In the pharmaceutical industry, new products pipelines are constantly changing as new products are identified and new treatment potentials are discovered. This research process requires several clinical trial phases which makes assessments difficult and it is subject to considerable uncertainty [54]. These uncertainties are both technological and market driven: any negative side effect can cancel a project and even if a project reaches FDA approval, a competitor's patent can also stop the project.

As in most management techniques, the best recommendations appear to be to plan ahead of time what to manage, how to manage it and most importantly what to do with it when it is time to retire it. The infrastructure industry has developed mature processes, which are recognized as international standards (IIMM) [55] and which provide detailed guidelines on how to plan and what to include in an asset management plan. This planning starts at the corporate level (i.e. strategic objectives) and requires that the sources of information are reviewed before working on establishing service levels and planning and executing a lifecycle management strategy. Considering the asset's lifecycle and financial forecasts, the plan is updated, reiterated and improved as required as indicated in **Figure 6**.

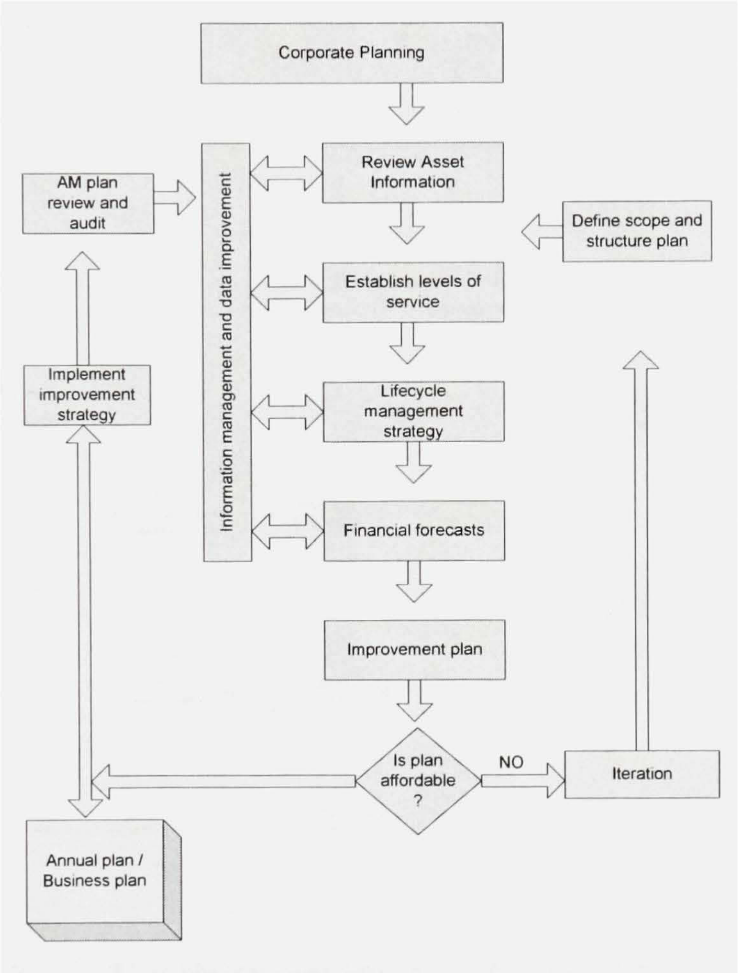

Corporate Planning
AM plan review and audit
Information management and data improvement
Review Asset Information
Define scope and structure plan
Establish levels of service
Implement improvement strategy
Lifecycle management strategy
Financial forecasts
Improvement plan
Is plan affordable ?
NO
Iteration
Annual plan / Business plan

Figure 6 Asset Management Plan - from IIMM [39]

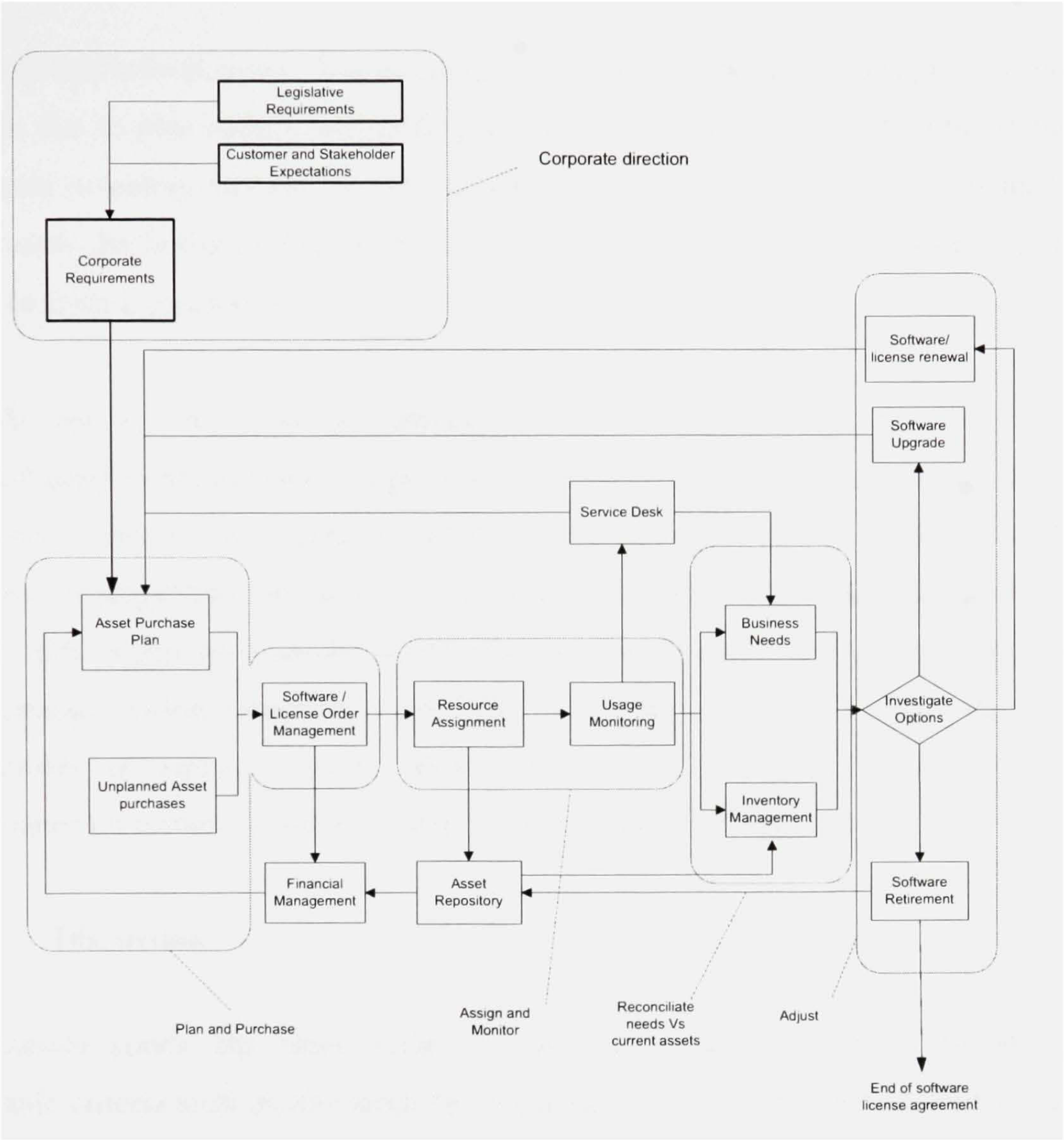


Figure 7 Combining concepts from ITIL and IIMM - from [56]

In **Figure 7** from [56], a high level SAM model is proposed, where corporate buy-in is first obtained; upper management also provides directives which help plan the SAM budget. This budget also receives input from the budget from the last financial period.

The purchasing group uses the budget to buy the planned software (and some unplanned software). This purchased software is assigned to a resource (human or hardware): this information must be updated in an asset repository. The usage of this software is monitored and information is sent to the service desk for reporting and support planning purposes (i.e. to plan budget and staffing to meet service levels). Based on the status of the current inventory and the business needs of the organization, a decision is made to either renew the license, retire the license or upgrade to a new version (from the same vendor or from a competitor).

Here, the Service Desk and the Financial Management process are those defined by ITIL; the intent is not to redefine a process that already exists but simply to interact with an existing process which already fulfills the desired goal. If the information about inventory management, software usage, software entitlements and business needs (current and future) were known at all time, this model could work without any more development; exiting processes from ITIL would be sufficient for the appropriate management of software assets. However, industry requirements and industries performance on managing software assets suggest otherwise (see section 1.2).

1.4 Discussion

For tangible goods, the replacement of physical goods is based on specific and observable criteria such as low stock or "wear and tear" of the material. For software, these criteria are not so clear cut: "Low stock" is not observable (i.e. a software can be copied; restrictions are not physical but legal). It also depends on several factors such as software usage and business needs (i.e. cost, functionality offered, strategic partnering issues) while respecting the terms and conditions of the software license (i.e. software entitlement); these terms and conditions often require manually intensive means to measure or evaluate. Furthermore, wear and tear do not apply to software, but limitations such as end of support from the software manufacturer and the balance between stability and new features are decisions that software users must tackle. All these factors must be understood and monitored to appropriately plan the SAM budget and its activities.

Further, management must be able to control software usage but oversight of software usage is often lost when end-users can install themselves software on their computers or from the internet or when they can change hardware without appropriate monitoring and control from the organization. This lost of oversight can even appear if too many patches or hardware upgrades are performed without constant control; any iteration can introduce a change to the software which may render the software unrecognizable from its original purchase form or state. Here technology is often asked to compensate: but without the equivalent of bar codes for software, this is rather a difficult task. Software identification, tracking and reconciliation to its purchasing order remain manually intensive for several organizations. To offset this uncertainty, organizations appear to overstock (i.e. overshooting the demand); furthermore, they do not benefit from the same tools and techniques as those found in supply chain management where stocks are kept low with the 'Just in time' technique, for example.

1.4.1 Research covered by literature

The literature review has identified topics that are common to several engineering fields as well as some specific to one engineering field. Regardless of the engineering fields, some are designed for intangible assets while others for tangible assets. **Table 7** summarizes these finding and classifies them according to whether they are specific to tangible assets or whether they are specific to intangible assets.

Table 7

Tangible asset vs. Intangible assets findings in the literature

Physical/ Tangible assets	Software / Intangible assets
1. Lot sizes are costly and need to be planned and monitored. 2. Conditions needs to be measured and monitored constantly at regular intervals 3. Wear and tear will be a key factor in determining when to replace an asset; these are measurable criteria. 4. A baseline needs to be established for a comparison base 5. Stocks can be costly and must be balanced against such things as volume discount and the value for the organization 6. Quality of control mechanisms can be evaluated and measured with statistical control techniques and tools 7. Some industrial engineering disciplines such as supply chain management have development tools and techniques to manage several asset simultaneously 8. Measures exist to determine wear and tear and other factor influencing when to replace and/or stop maintenance on an asset. 9. With defined measures and sets of criteria, automation of measurement is possible. 10. Supply management covers the exchange of intermediate goods between organizations and considers the fact the information can be imperfect. 11. Various functional goals will lead to diverging interests in terms of inventory management goals 12. Four (4) factors explain the existence of inventory: Time to develop the product, discontinuity on the production process, uncertainties and volume discounts.	1. The industry is concerned with the increasing cost of managing software assets 2. ITIL is a de facto point of reference for several software tool vendors 3. ITIL is a good start but does not provide details on how to manage software assets and/or license entitlements. 4. Commercial software are licensed to buyers with some restrictions: the terms and conditions. The lack of process and techniques from license owner on how to ensure entitlements, makes this task a very manually intensive one. 5. Outsourcing and other contractual agreements force organizations to remove ambiguity but terms and conditions of software license remain hard to define and manage (i.e. manually intensive) 6. Software asset management is not well defined in the industry and has led the industry to provide several competing solutions.

1.4.2 Research topics not covered in the literature

Some of the findings that apply to physical (tangible) assets could also be applied to software (intangible) assets, Table 8 lists some of the physical findings that can be applied to software assets and vice-versa. In the tangible column (physical goods), some researchers mention the simplicity of some models and how they could be improved. In intangible goods column, the lack of data, measures and general understanding on how SAM impact the business is mentioned.

Table 8

Tangible vs. Intangible goods: items not addressed in the literature

Physical/ Tangible goods	Software / Intangible goods
1. In supply-chain management; most models consider a single supplier	1. Research does not cover well software assets (usage and control of software that have terms and conditions as legal constraints); it focuses more on the construction of software 2. Observable criteria and decision factors that influence positively or negatively SAM are not discussed in research articles (only in industry white papers). 3. Indicators and measures that indicate how efficient licenses are used are not yet discussed with software assets. 4. Apart from ITIL, they are no references to industry frameworks to help define what is SAM and how to appropriately manage it. 5. Predefined control points: factors that can be measured and can be used to determine when to replace or stop maintenance of a software. 6. The definition of software usage and the automation of the monitoring of software compliance 7. The reasons for the existence of inventory is not well documented in the IS/IT industry.

CHAPTER 2

RESEARCH METHODOLOGY

2.1 Research objectives

To address this industry problem in software asset management, this thesis focuses on the definition of what is included in the scope of Software Asset Management (SAM) and by doing so, aims to construct a common vocabulary to describe SAM. The specific objectives of the thesis are :

1. Actively contribute to the development and to the content of the ISO international standard on SAM (ISO/IEC 19770-1).
2. Capture, identify and analyze elements that are relevant to SAM, including those that would not make it into the final version of the international standard.
3. Provide an analysis of the international SAM standard with respect to the 27 processes within ISO/IEC 19770-1.
4. Develop an exploratory assessment method to allow organizations to determine their gaps against ISO/IEC 19770-1.

2.2 Research Methodology

To address the research objectives, the requirements for the construction of a set of SAM processes have to be identified while taking as inputs existing software models and techniques from related knowledge domains - including software engineering:

1. Define criteria to assess how well an organization manages its software assets, taking into account the concerns and knowledge of the industry and the research community.
2. Construct a model (e.g. a set of processes) to define the processes that are required to perform SAM
3. Through a panel of experts, verify the SAM model and develop a SAM standard

4. Define the assessment rules that will enable the processes of the model to be evaluated.
5. Provide an analysis of the SAM model to better interpret the assessment results.
6. Provide a representation of the result of the assessment method to help identify risk areas for the organization.
7. Provide some indications of how the proposed SAM model and its assessment can impact the industry and the research community.

The definition of the objectives are the first step of the methodology which is borrowed from a model that was designed for the construction of a measurement method [57]. The model in [57] had to be modified since the goal here is not to measure software but to assess how well an organization manages its software assets. The modified research methodology model is represented in Figure 8 and detailed in the following sections.

2.3 Step 1: Design of the assessment method

As described in [57], the first substep is to specify what we want to assess; in this case, it is how well an organization performs SAM. If the organization does not perform well, it should be possible to identify why and what portion of SAM is not well performed.

This implies that we know what we are looking for during the assessment and that there are two points of reference; one to determine what is SAM and another one to rate organizations in their capability to perform SAM. This leads to the following substeps; substep 2 defines the characteristics that will be used to determine the conformity to a specific process. Substep 3 is to specify what assessment concept is being tracked: specific attributes should be defined. In this context, the capability of an organization to carry out a process is being assessed.

Substep 4 requires the definition of the assessment numerical assignment rules. If a capability maturity model such as CMMI [33]or ISO/IEC 15504 [58] is used, then all these substeps are defined since each of the assessment standards has a rating of the maturity of a process (e.g. from incomplete – level 0 to optimizing – level 5 in the ISO/IEC 15504 process assessment reference model).

2.4 Step 2: Assessment method application

Once the assessment is defined, it should be possible to apply it to the desired model. However, in this case, the model has to be constructed. The first substep of step 2 is thus to gather all the relevant processes in order to have outcomes that can be assessed against the assessment method of step 1. The second subset of step 2 is to construction the SAM model that will be assessed. The construction of the processes is strongly influenced by existing standards (ITIL, BS15000) and by ISO/IEC 19770-1 developed in parallel and described in CHAPTER 3 .

In parallel to the construction of the SAM model, an initiative from ISO is being conducted and is being used to verify the model being constructed. The verification is described in CHAPTER 4 : a historical approach is taken by documenting the evolution of the documents produced during the construction of ISO/IEC 19770-1, from the first draft until the publication of the standard itself. More specifically, four document versions are examined. From the first draft to the final standard, changes in each version are discussed and decisions to include or exclude topics are also noted and analyzed.

By taking a historical approach to the development of ISO/IEC 19770-1 on SAM processes it is possible to examine which topics were discussed and which ones made a consensus and eventually made it into the final version of the standard. The approach used for this thesis was to get nominated in 2003 at the beginning of this PhD project as the Canadian representative: this provided me with the opportunity to have access to a panel of experts and it served as well as a sounding board to test new ideas. This panel of experts is also used to verify which processes have an industry consensus (through ISO's approbation structure) and which ones do not yet have the industry consensus.

Substep 3 of the step 2 is to assign a numerical value to the assessment methodology: this corresponds to the assessment of the level of maturity of the SAM model as described in CHAPTER 5 .

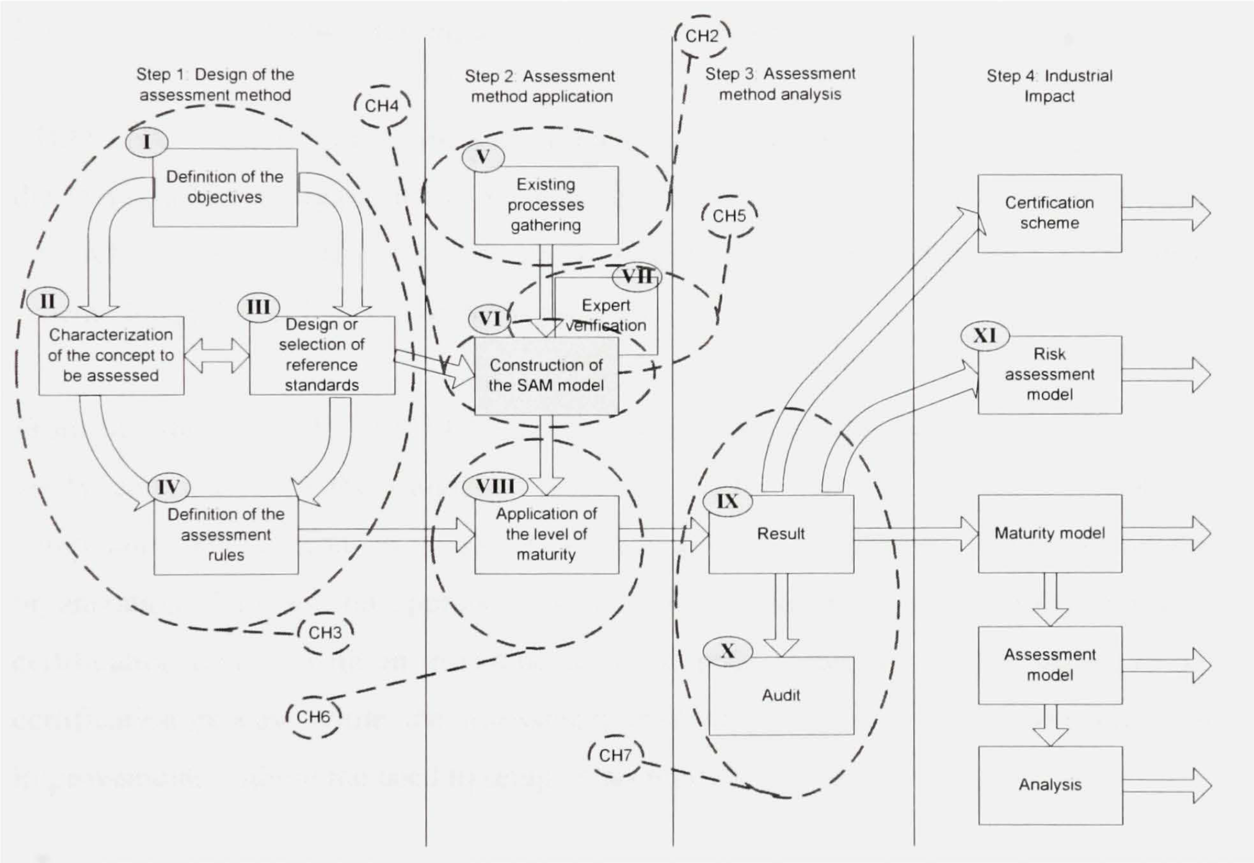

Step 1 Design of the assessment method
Step 2 Assessment method application
Step 3 Assessment method analysis
Step 4 Industrial Impact
CH2
CH4
CH5
CH3
CH6
CH7
I
Definition of the objectives
II
Characterization of the concept to be assessed
III
Design or selection of reference standards
IV
Definition of the assessment rules
V
Existing processes gathering
VI
VII
Expert verification
Construction of the SAM model
VIII
Application of the level of maturity
IX
Result
X
Audit
XI
Certification scheme
Risk assessment model
Maturity model
Assessment model
Analysis

Figure 8 The SAM research methodology

2.5 Step 3: Assessment method analysis

In [57], two substeps are presented; substep 1 is the presentation of the assessment and evaluation of the results and substep 2 is an audit to ascertain the quality of the assessment. In this case, the audit is the assessment; in CHAPTER 6 , ISO/IEC 19770-1 is presented and analyzed in terms of control points and how it relates to the organization and the IT infrastructure. This analysis allows for a better understanding of the result of the assessment and its impact on the organization; not knowing where software assets intervene in the organization is a risk; even if the organization knows where to focus, the rigor and formalism to manage SA will also influence the level of risks the organization is taking in SA management.

2.6 Step 4: Industrial impact of standardization

CHAPTER 8 describes how the standard and the ongoing research has and will impact the IS/IT industry; organizations are getting involved in the definition of the ongoing research on the SAM tag (ISO/IEC 19770-2) and on the construction of a certification scheme for ISO/IEC 19770-1 on SAM processes

From the start, ISO/IEC 19770-1 has been designed with the intent of being used to verify compliance to the standard. However, another approach is not to check for compliance but instead to assess how mature each process is within a specific organization. This second approach has several consequences: compliance implies a certification process with an infrastructure to accredit organizations that adhere to the certification process while the assessment method can provide recommendations of improvements without the need to setup an accreditation scheme.

CHAPTER 3

CONSTRUCTION OF THE SAM MODEL

3.1 Objectives of the SAM model

In this chapter, findings from the literature review are used to ensure that all important topics will be covered by the proposed set of processes to define the content and scope of SAM. In parallel, existing standards such as ITIL, BS15000 and the development work for ISO/IEC 19770-1 on SAM processes are used as inputs. The objective of this chapter is to present the construction of a SAM model (e.g. a set of processes) to identify and define the processes that are included in SAM. This SAM model is a necessary input for the assessment method described in CHAPTER 2.

3.2 Structuring the literature review findings

The findings of the literature review can be reworded as criteria to take into account the concerns and knowledge of the industry and of the research community:

1. Planning must start at the corporate level.
2. The number of software, or lot size, must be planned and monitored using inventory management practices and techniques.
3. The state or condition of the product must be monitored and measured (such as wear and tear) when possible.
4. A baseline must be established to evaluate how the inventory changes overtime.
5. Statistical control techniques are used to measure quality control.

6. Choices must be made about an equilibrium between cost vs. quality and stability vs. new functionality: simulations are a good technique to evaluate the impact on the IT infrastructure.
7. Automation is necessary to manage large number of information but it requires that processes be defined for it to be effective.
8. To determine the adequate stock levels, information across the organization or across organizations need to be current and adequate.
9. The goals of SAM need to be balanced out against other functional goals within the organization, each having diverging interests in terms of inventory management goals.
10. There is a need to make the terms and conditions included with commercial software more explicit as to simplify the process of determining if these terms are met or not (i.e. a very manually intensive and non reproducible process implies an undefined process).
11. There is a need to clarify what encompasses SAM related activities especially in outsourcing and other contractual agreements where ambiguity remains an issue.
12. There is a need for a single point of reference to reduce the number competing solutions and approaches to SAM.
13. There is a need to determine the factors that influence positively or negatively SAM; one such factor could be used to determine when to replace or stop maintenance of a software.
14. There is a need to define and described the reasons for the existence of inadequate levels of software stocks (i.e. under or over licensing) as it exist for physical assets.

The next step requires to translate the criteria listed above into requirements: each of those requirements could possibly become the goal of a process. To simplify the process of transforming the criteria into requirements, the criteria are grouped according to certain communalities of goals and purpose. The next section examines existing grouping of processes and provides an indication on how to group the criteria and requirements.

3.3 Positioning SAM with respect to existing standards

Figure 9 shows that SAM can be positioned with respect to existing standards such as ITIL, BS1500 and ISO/IEC12207. The first thing to notice is that any software will need to be introduced and implemented in the IT infrastructure. This means that the IT infrastructure is the object to be managed and monitored. **Figure 9** also indicates that ITIL is closely linked to the IT infrastructure; ITIL has thus a very operational focus. This means that for operational concerns, ITIL can be a good point of reference. However some of the criteria listed in section 3.2 refer to managerial criteria and concerns: other points of reference are necessary.

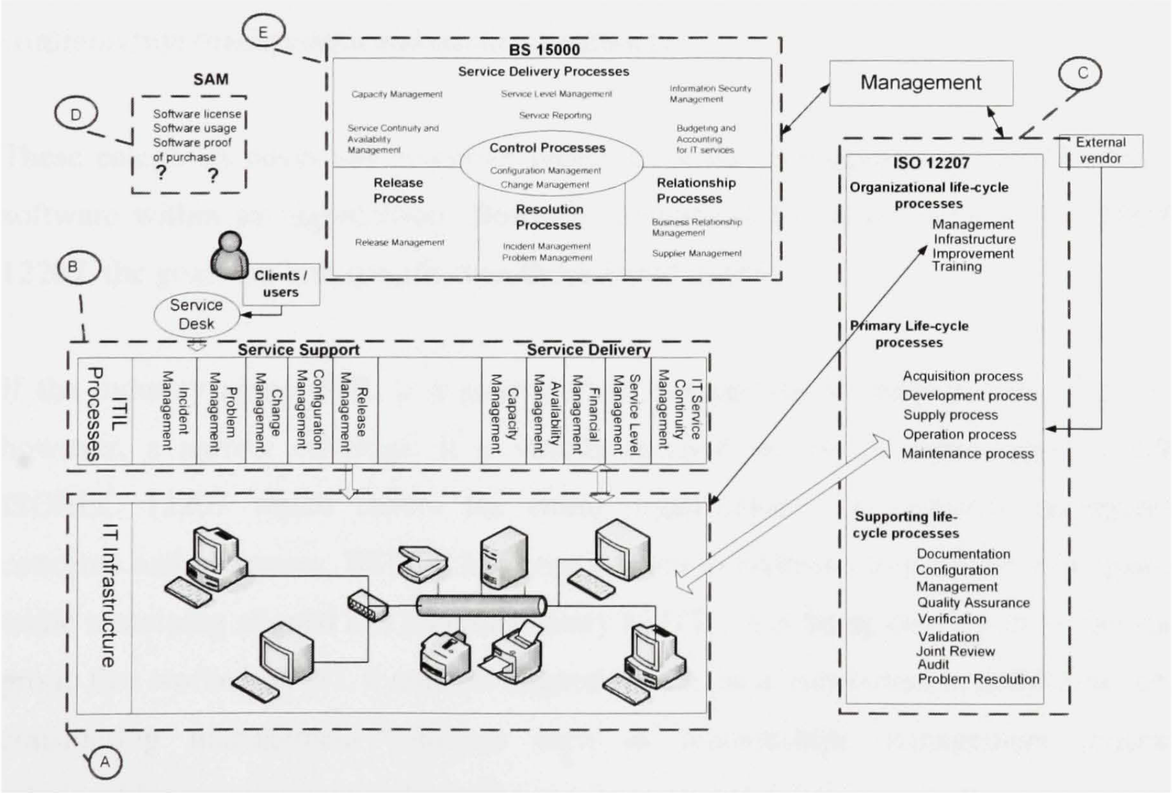


Figure 9 Processes and infrastructure

Figure 9 also refers to ISO/IEC 12207 which has 3 categories of processes. One of these 3 categories, the "Primary life-cycle processes", contains processes for the IT infrastructure. The processes of the primary life-cycle processes group start with the acquisition and plan for the operating and maintenance of the software; however, as in ITIL, there is no mention of what to do about licenses and how to manage assets while ensuring compliance to the terms and conditions.

In ISO/IEC 12207, there are also two other groups of processes: the Organizational processes and the Supporting processes. Organizational processes focus on processes that are not specific to the IT infrastructure such as management, training and improvement processes; infrastructure processes are also included in this category as a separate process to take into account any process needed to manage the IT infrastructure but without any explanation on where it is applied. Supporting processes are processes that can be used by other processes to help use other processes, such as: documentation, configuration management and quality assurance.

These categories cover any processes required for the management of the life cycle of software within an organization. However, because of the large coverage of ISO/IEC 12207, the goals are less specific than those found in ITIL.

If the industry agrees ITIL is a good start to manage the IT infrastructure, ITIL has, however, a narrow coverage: it is strictly focused on the IT infrastructure (unlike ISO/IEC 12207 which covers the entire organization). To consider management concerns and processes, BS1500 has been created to address service level management while remaining aligned and complementary to ITIL: it is being developed by the same group that worked on ITI. It remains aligned to ITIL as it uses common definitions while considering management concerns such as relationships management (business relationships management and supplier management) and service reporting.

To ensure that processes cover the entire organization, including management and the IT infrastructure, the structure of ISO/IEC 12207 will be borrowed (i.e. organization level, Infrastructure level (Core SAM processes) and interfacing/support processes level). However, to be more focused on the IT infrastructure, the granularity of ITIL and BS15000 will be used when relevant.

3.3.1 Process grouping

To help build the processes required to manage SAM, the criteria listed in section 3.2 are categorized into the 3 categories of processes: organization level, infrastructure SAM level and Interface processes level.

Table 9

Criteria associated to a process group

#	Need	Group
01	Planning	Org (planning)
06	Infrastructure Choice	Org (Infrastructure)
08	Reporting information	Org (reporting)
07	Automation	Org (improvement)
09	SAM goals	Org (planning)
14	Stock management	Org (all)
10	Define terms and conditions	SAM (Compliance)
11	Define SAM scope	SAM (all)
12	SAM standardization	SAM (all)
13	SAM planning and monitoring criteria	SAM (all)
02	Acquisition	Interface (acquisition)
03	State indicator	Interface (Configuration)
04	Baseline	Interface (Configuration)
05	Quality Control	Interface (Quality)

3.3.2 Organizational requirements

Organizational requirements address concerns that apply to the entire organization often with a link to upper management. Table 10 presents the processes taken from the organizational criteria.

Table 10

Organizational criteria

#	Process label	Mapping	Source
01	Planning	Planning	ISO 12207
06	Infrastructure Choice	Infrastructure	ISO 12207
08	Reporting information	Service reporting	BS 15000
07	Automation	Improvement	ISO 12207
09	SAM goals	Planning	ISO 12207

3.3.2.1 The granularity of processes

Before mapping the processes to the relevant standards identified above, it is important to notice an important variation in the level of granularity between ITIL/BS15000 and ISO 12207:

- For ISO/IEC 12207, organizational processes are generic and expressed at a very high level.
- For ITIL/BS15000, processes are at a much lower level of granularity, closer to the IT infrastructure.

For example, the single process "Infrastructure" (i.e. "a process to establish and maintain the infrastructure needed for any other process") of ISO/IEC 12207 encompasses all of ITIL's processes.

This means that processes that are not organizational will have a level of granularity closer to ITIL/BS15000 while organizational processes will have a granularity closer to ISO/IEC 12207.

3.3.3 Resulting SAM model

The research phase consists in mapping existing standards to the three groups identified earlier (organizational, SAM, interface) and examine how the new set of processes relate to the existing processes. The details of this mapping exercise can be found in APPENDIX VIII while Figure 10 depicts the resulting SAM model. In parallel to this exercise, ISO/IEC 19770-1 on SAM processes was being constructed and also covers the same scope and adheres to the same principles: this alignment is to be expected since my involvement with the development of this ISO standard ensures an alignment between the two initiatives. There are, however, some differences: CHAPTER 4 provides more details on the development of ISO/IEC 19770-1 while CHAPTER 6 provides more details providing an analysis of the SAM model and discusses how it relates to management or the IT infrastructure activities.

By looking at Figure 10, it is important to notice that ISO/IEC 19770-1 covers most of the SAM processes relevant to SAM identified within this chapter. However, in this proposed SAM model configuration management, service reporting, audit and assessment and entitlement management found in other processes such as BS15000 are not present in ISO/IEC 19770-1 even though the literature review suggests they are relevant to the management of software assets (indicated in red in figure 10).

Nevertheless, it is important to notice the strong influence of BS15000 (now ISO/IEC 20000) on the content of ISO/IEC 19770-1: all BS15000 processes that have found their counterpart in ISO/IEC 19770-1 are highlighted by a blue frame in Figure 10. There are eight such processes.

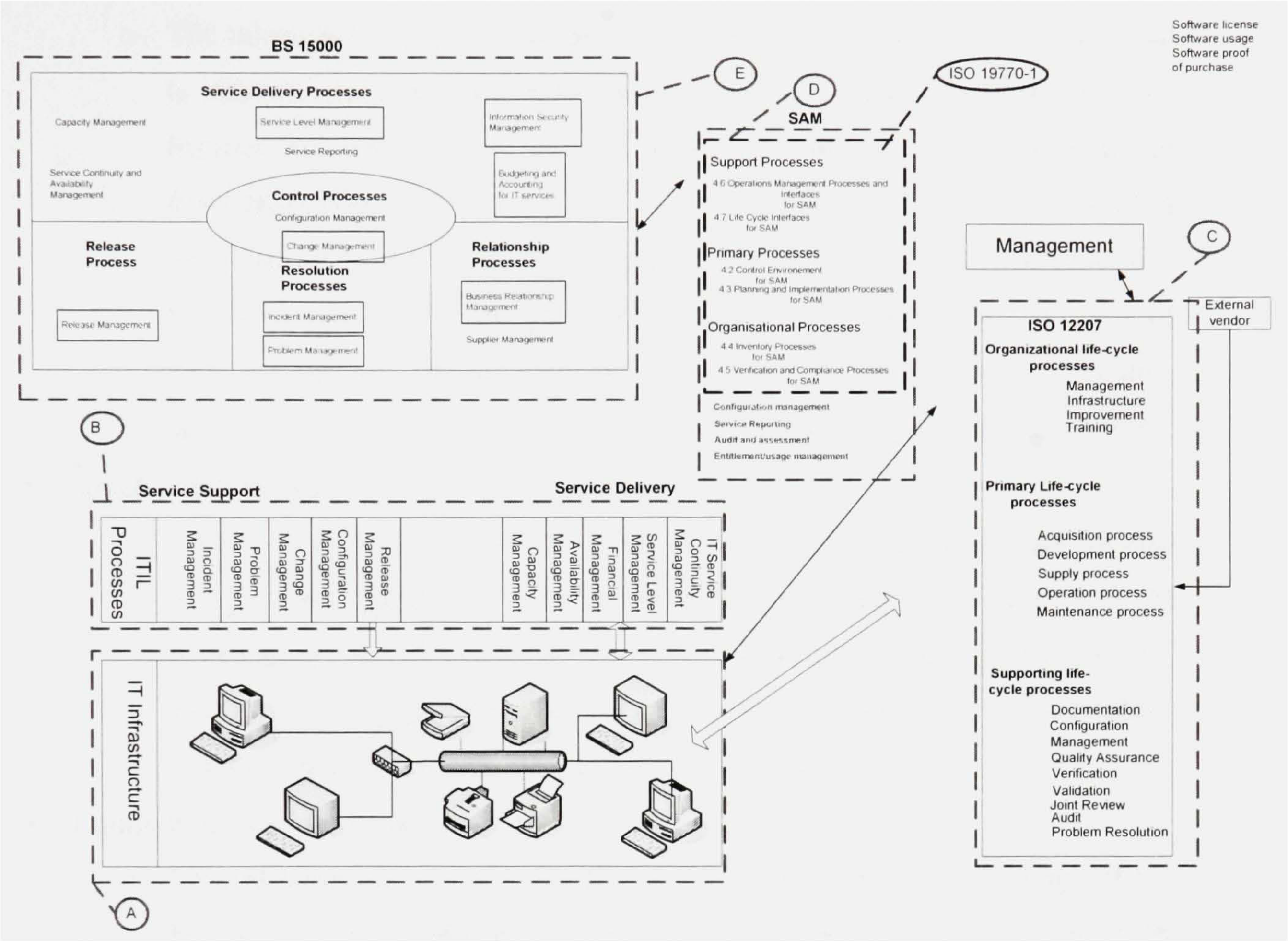


Figure 10 Positioning the ISO based SAM processes

Even though ISO/IEC 19770-1 covers most of the SAM related processes, the topics indicated in red in Figure 10 are not present in this ISO standard. However, the content can be relevant to the management of SAM;

- Configuration management:
    - This process is found in ITIL, BS15000 and ISO/IEC 12207 to account for all the IT assets and configurations within the organization.
    - As part of this process, a baseline is established to be able to tack any changes to the original configuration of the IT infrastructure.
- Service reporting

  - The information collected has to be gathered and presented in a manner that is relevant and useful to the organization. This requires knowledge of how licenses and the breach of contracts can impact the organization in terms of fines and damage to the reputation of the organization. Furthermore, a SAM plan must be produced yearly as part of ISO/IEC 19770-1 set of processes but there is no detail on where the information comes from. The goal of service reporting happens to be: "To produce agreed, timely, reliable, accurate reports for informed decision making and effective communication"
- Audit and assessment
  - There is a need to determine the compliance with the SAM processes but also to assess the characteristic of the organization in order to construct a SAM plan that is relevant to the needs and requirements of the organization. By assessing the organizations level of maturity, it is easier to determine where to focus; this information can be part of the SAM plan.
- Entitlement or usage management
  - Several terms and conditions of licenses are dependent on the usage of the licensed software. However, there are no automated means provided by software manufacturer to monitor or report on the usage of several commercial software: this means that verification of compliance to licensing terms is difficult at best. It is thus not clear what must be monitored and tracked (or measured when possible) in order for compliance reporting to be valid for both parties: the licensor and the licensee.

The construction of the ISO/IEC 19770-1 and the details of the structure are detailed next in CHAPTER 4 .

CHAPTER 4

ISO/IEC 19770-1 DEVELOPMENT

4.1 Context : Origins and motivation for ISO/IEC 19770-1

The year 2000 brought a lot of activity in the IS/IT industry. The practice of using only 2 digits to represent the year made sense in the 1950s, 1960s and 1970s as the cost of storage and CPU usage was fairly important. But as time passed, the cost of storage and CPU usage lowered and when the year 2000 was approaching, this practice caused an interpretation problem: with only two digits, it would be impossible to distinguish between the year 1900 and the year 2000, the year 1901 and 2001, and so on. This meant that all software applications and computer programs had to be fixed to avoid this confusion: this became known as the millennium bug.

It also meant that organizations had to perform a rigorous and exhaustive inventory of all the software applications being used by the organization. This task was complex and complicated as there was no indication on how to perform a software asset inventory and there was also no standardized way of locating and identifying all software assets owned and used by an organization.

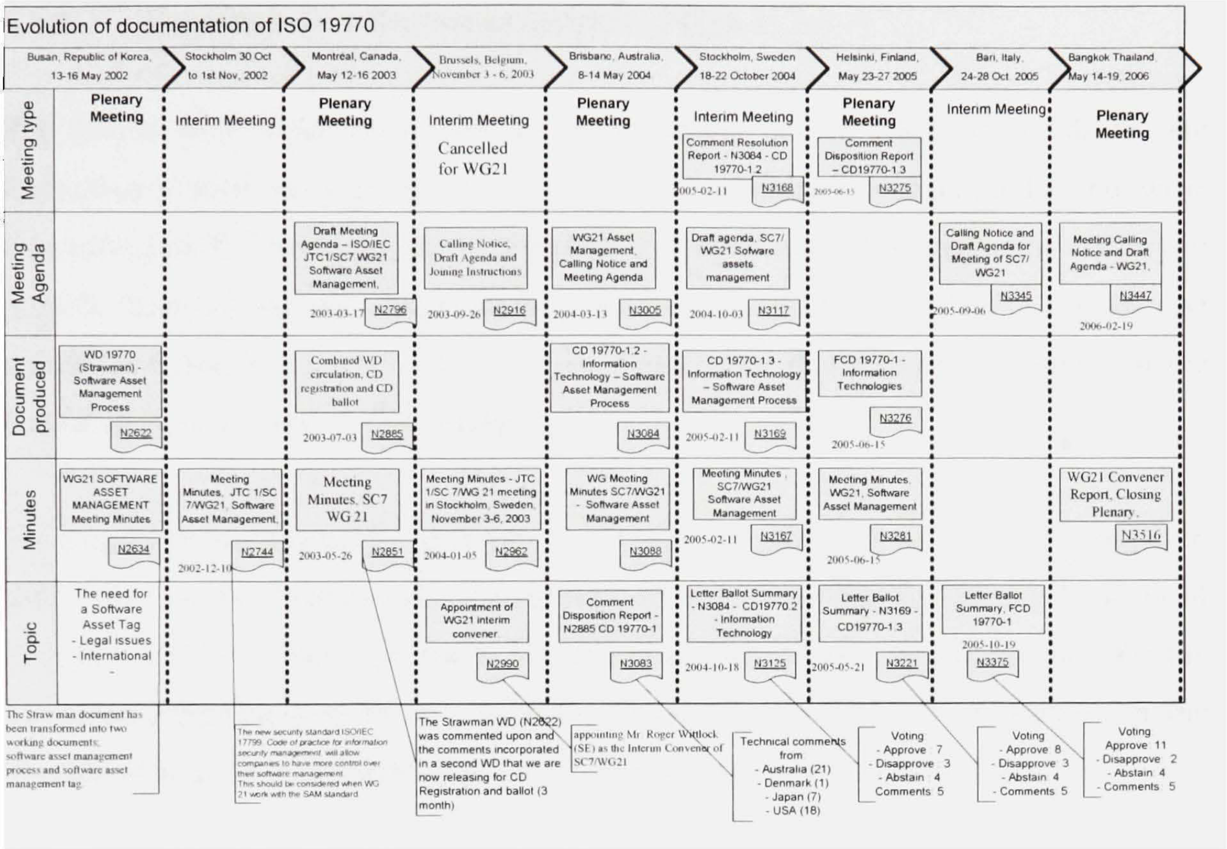
After the year 2000 and after the millennium bug had passed, it was evident that obtaining a complete and exact software asset inventory was a very difficult task and the lack of standards made it difficult to perform it in an efficient and repeatable manner. A group of Swedish organizations decided to propose an international standard. In 2001, the Swedish standard institute (SIS) asked ISO to form a working group to address this issue: the ISO/IEC JTC1/SC7 Working Group 21 (WG21) was formed.

This chapter describes the process that led to the creation of ISO/IEC 19770-1 SAM processes. From Figure 11, it can be seen that the development of ISO/IEC 19770-1 started in 2001 and the final version of the document was published in 2006 as an International Standard. During that five year period, the content of this standard has changed and evolved as different topics and concerns were discussed and voted on by the international community of experts in SAM, in which I contributed significantly throughout this thesis project. More detailed about the evolution of the ISO/IEC 19770-1 standard in provided in APPENDIX X.

4.2 A chronological view of ISO/IEC 19770-1

Figure 11 shows that each document produced within ISO/SC7 is assigned an unique identifier regardless of whether it is a draft document, meeting minutes/agenda or a resolution. For example, the drafts of the standard that are covered in the following section have the following identifiers: N2622 for the Busan version of the document produced in 2002, N2885 for the Montreal version of the document produced in 2003, N3084 for the Brisbane version of the document produced in 2004 and N3276 for the Helsinki version of the document produced in 2005.

All the official documents produced by ISO's Joint Technical committee 1/Sub Committee 7 (JTC1/SC7) are attributed a unique identifier and can be found on their web site (www.jtc1-sc7.org) ordered according to their identifiers. However, those documents can refer to any standard in development and can be of any document type. Figure 11 presents only the documents that relates to ISO/IEC 19770-1 and they are grouped according to their document type: meeting agenda, working document produced, meetings and other topics.



Evolution of documentation of ISO 19770

	Busan, Republic of Korea, 13-16 May 2002	Stockholm 30 Oct to 1st Nov, 2002	Montreal, Canada, May 12-16 2003	Brussels, Belgium, November 3 - 6, 2003	Brisbane, Australia, 8-14 May 2004	Stockholm, Sweden 18-22 October 2004	Helsinki, Finland, May 23-27 2005	Bari, Italy, 24-28 Oct 2005	Bangkok Thailand May 14-19, 2006
Meeting type	Plenary Meeting	Interim Meeting	Plenary Meeting	Interim Meeting Cancelled for WG21	Plenary Meeting	Interim Meeting Comment Resolution Report - N3084 - CD 19770-1.2 2005-02-11 N3168	Plenary Meeting Comment Disposition Report – CD19770-1.3 2005-06-15 N3275	Interim Meeting	Plenary Meeting
Meeting Agenda			Draft Meeting Agenda – ISO/IEC JTC1/SC7 WG21 Software Asset Management, 2003-03-17 N2796	Calling Notice, Draft Agenda and Joining Instructions 2003-09-26 N2916	WG21 Asset Management, Calling Notice and Meeting Agenda 2004-03-13 N3005	Draft agenda, SC7/ WG21 Sofware assets management 2004-10-03 N3117		Calling Notice and Draft Agenda for Meeting of SC7/ WG21 2005-09-06 N3345	Meeting Calling Notice and Draft Agenda - WG21, N3447 2006-02-19
Document produced	WD 19770 (Strawman) - Software Asset Management Process N2622		Combined WD circulation, CD registration and CD ballot 2003-07-03 N2885		CD 19770-1.2 - Information Technology – Software Asset Management Process N3084	CD 19770-1.3 - Information Technology – Software Asset Management Process 2005-02-11 N3169	FCD 19770-1 - Information Technologies N3276 2005-06-15		
Minutes	WG21 SOFTWARE ASSET MANAGEMENT Meeting Minutes N2634	Meeting Minutes, JTC 1/SC 7/WG21, Software Asset Management, 2002-12-10 N2744	Meeting Minutes, SC7 WG 21 2003-05-26 N2851	Meeting Minutes - JTC 1/SC 7/WG 21 meeting in Stockholm, Sweden, November 3-6, 2003 2004-01-05 N2962	WG Meeting Minutes SC7/WG21 - Software Asset Management N3088	Meeting Minutes , SC7/WG21 Software Asset Management 2005-02-11 N3167	Meeting Minutes, WG21, Software Asset Management N3281 2005-06-15		WG21 Convener Report, Closing Plenary. N3516
Topic	The need for a Software Asset Tag - Legal issues - International -			Appointment of WG21 interim convener N2990	Comment Disposition Report - N2885 CD 19770-1 N3083	Letter Ballot Summary - N3084 - CD19770.2 - Information Technology 2004-10-18 N3125	Letter Ballot Summary - N3169 - CD19770-1.3 2005-05-21 N3221	Letter Ballot Summary, FCD 19770-1 2005-10-19 N3375	

The Straw man document has been transformed into two working documents: software asset management process and software asset management tag

The new security standard ISO/IEC 17799: *Code of practice for information security management*, will allow companies to have more control over their software management. This should be considered when WG 21 work with the SAM standard

The Strawman WD (N2622) was commented upon and the comments incorporated in a second WD that we are now releasing for CD Registration and ballot (3 month)

appointing Mr. Roger Wittlock (SE) as the Interim Convener of SC7/WG21

Technical comments from
- Australia (21)
- Denmark (1)
- Japan (7)
- USA (18)

Voting
- Approve 7
- Disapprove 3
- Abstain 4
Comments 5

Voting
- Approve 8
- Disapprove 3
- Abstain 4
- Comments 5

Voting
Approve 11
- Disapprove 2
- Abstain 4
- Comments 5

Figure 11 ISO/IEC 19770-1's evolution within ISO/SC7

4.3 The evolution of ISO/IEC 19770-1 through its documents

4.3.1 SAM processes and TAG

As indicated in Figure 11, in 2001 a working group, Working Group 21 (WG21), started to work on defining a standard for software asset management. At that time, WG21 was more concerned about defining a SAM tag than it was about defining SAM processes as it was driven by market concerns about improving the accuracy of the inventory and the discovery tools used for that purpose.

4.3.2 Canadian contribution to ISO/IEC 19770-1

The formal ISO organization acts like a publishing house: it sets standards for the formatting of texts and graphics as well as the level of English used, but it does not write the content itself. For the technical expertise, ISO relies on each participating country to provide the expertise and manpower to write the content of standards. For Canada, the Standards Council of Canada (SCC) is the organization that provides resources for the Canadian contributions to ISO standards.

The implication of the author in the development of the standard with ISO/SC7 began in 2003 when the SCC submitted my name to participate in WG21 for the development of ISO/SC7 19770 on Software Asset Management. The author then became in 2003 the Canadian representative for the standard; being the first and only Canadian on the Working Group; it is not until 2006 that another Canadian joined the group.

As indicated in Figure 11, WG21 was still working on a draft version of the document in 2003. In the ISO standards development process, in order to move up in stages, documents produced must be voted upon and these documents change status only during the yearly plenary meeting which is held in May of every year.

The evolution of the standard will be covered through the evolution of the documents presented in each plenary meeting; comments provided before the plenary meetings and the interim meetings will also be presented when these comments provide additional relevant information. This controlled evolution process acts as a verification and validation process to ensure the quality of the content of ISO documents, while building an industry consensus at an international level.

4.3.3 The Busan-2002 (strawman) version (N2622)

4.3.3.1 Purpose

In the Busan version of the standard (N2622) produced in 2002, the purpose of the standard was "to meet market needs/requirements to reduce costs, keeping a good software management and taking control of systems and software assets."

It is important to notice that from the start, the standard was driven by market demands to reduce costs and control SAM. However, before specifying how to achieve this, it was important to assess the current state of the industry.

4.3.4 The Montreal-2003 version (N2885)

4.3.4.1 Purpose

In the Montreal version of the standard (N2885) produced in 2003, the purpose of the standard is to "meet market needs/requirements to reduce costs, take control of systems and software assets and thus maintain status of software license compliance. It also provides the related technical guidance document that is required to support the standard".

This purpose has been changed from the previous version; it now specifies that the organization must maintain the status of the license compliance and provide technical guidance to support the standard. These are two new requirements: one for dealing and managing the information about software compliance and the other about providing technical guidance to support the standard.

4.3.4.2 Requirements for SAM processes

Three other processes are introduced: order management, inventory management and license management which are used to secure the proof of purchase, demonstrate control over installed software and secure licensing documentation respectively. Furthermore, this version of the standard recognizes that the consolidation of all these elements has to be performed in order to determine the level of compliance. Table 11 presents the new topics that have been introduced with the Montreal version, but also the ones that have been dropped such as the interface with the help desk and change management.

Table 11

Variations from N622 to N2885

New topics	Topics dropped
1. Global and local organizations 2. Software usage guide 3. Proof consolidation process 4. The notion that the level of compliance must be sustained. 5. Technical support must be provided	1. Interface to help desk 2. Interface to change management 3. Merger and de-merger of organization 4. Licensing complexity 5. Standard Application catalogue 6. Different Asset Management roles

4.3.5 The Brisbane-2004 version (N3084)

4.3.5.1 Purpose

In the Brisbane version of the standard (N3084) produced in 2004, the purpose of the standard is to "establish a common framework for implementing and maintaining effective Software Asset Management Processes, in order to meet market needs/requirements to reduce costs, take control of systems and software assets and manage software asset license compliance"

The purpose of the standard was reworded in this version to specify that the standard is there to implement and maintain effective SAM; the notion of maintaining the standard has been moved at the beginning of the paragraph, increasing its importance.

Table 12

Variations from N2885 to N3084

New topics	Topics dropped
1. More emphasis on the maintenance of the process 2. Communication of the processes 3. Resolution of discrepancies between physical assets and licenses 4. The notion that the level of compliance must be sustained. 5. The disposal of software and its licensing documentation. 6. A guide to SAM that addresses audits, scanning processes, licensing types and security 7. Software may be removed from the infrastructure (Software asset retirement) but still available to the organization through release and installment. 8. Once the license is removed (Software Asset removal), the software may not be installed anymore without breaking the terms and conditions of the license. 9. An asset repository contains both the physical software and the license	1. Interface to help desk 2. Interface to change management 3. Merger and de-merger or organization 4. Licensing complexity 5. Standard Application catalogue 6. Different Asset Management roles

4.3.6 The Helsinki-2005 version (N3276)

4.3.6.1 Purpose

In the Helsinki version of the standard (N3276) produced in 2005, the purpose of the standard is to "establish a common basis for assessing whether an organization has successfully implemented to a baseline standard; an integrated set of processes for software asset management (SAM)"

This version of the purpose statement removes references about it being driven by the market (although it remains so). The term framework has also been replaced by "a baseline standard; an integrated set of processes"

In this version of the standard, more effort is spent selling the benefits of using the standard. A section is dedicated to the benefits that an organization can obtain if it applies and uses the standard: compliance to this standard should achieve some benefits in terms of risks management, costs control and competitive advantage. The Helsinki plenary meeting is also the first meeting where a British delegation that participated in the writing of ITL is present. So, it is no coincidence that the benefits cited are the same as those listed in the SAM book [19] from ITIL.

4.3.6.2 Requirements for SAM processes

In the Helsinki version of the standard, several changes and additions were introduced when some of the UK delegates who joined WG21 in January 2005 came with their ITIL expertise: this input influenced the content and structure of the document. More specifically, several of the changes proposed by the UK delegates were aligned with ITIL but also with service level management processes of BS1500 (being transformed into ISO/IEC 20000). The scope of the proposed changes was significant since the structure (i.e. the grouping of processes) and the labeling of some processes changed.

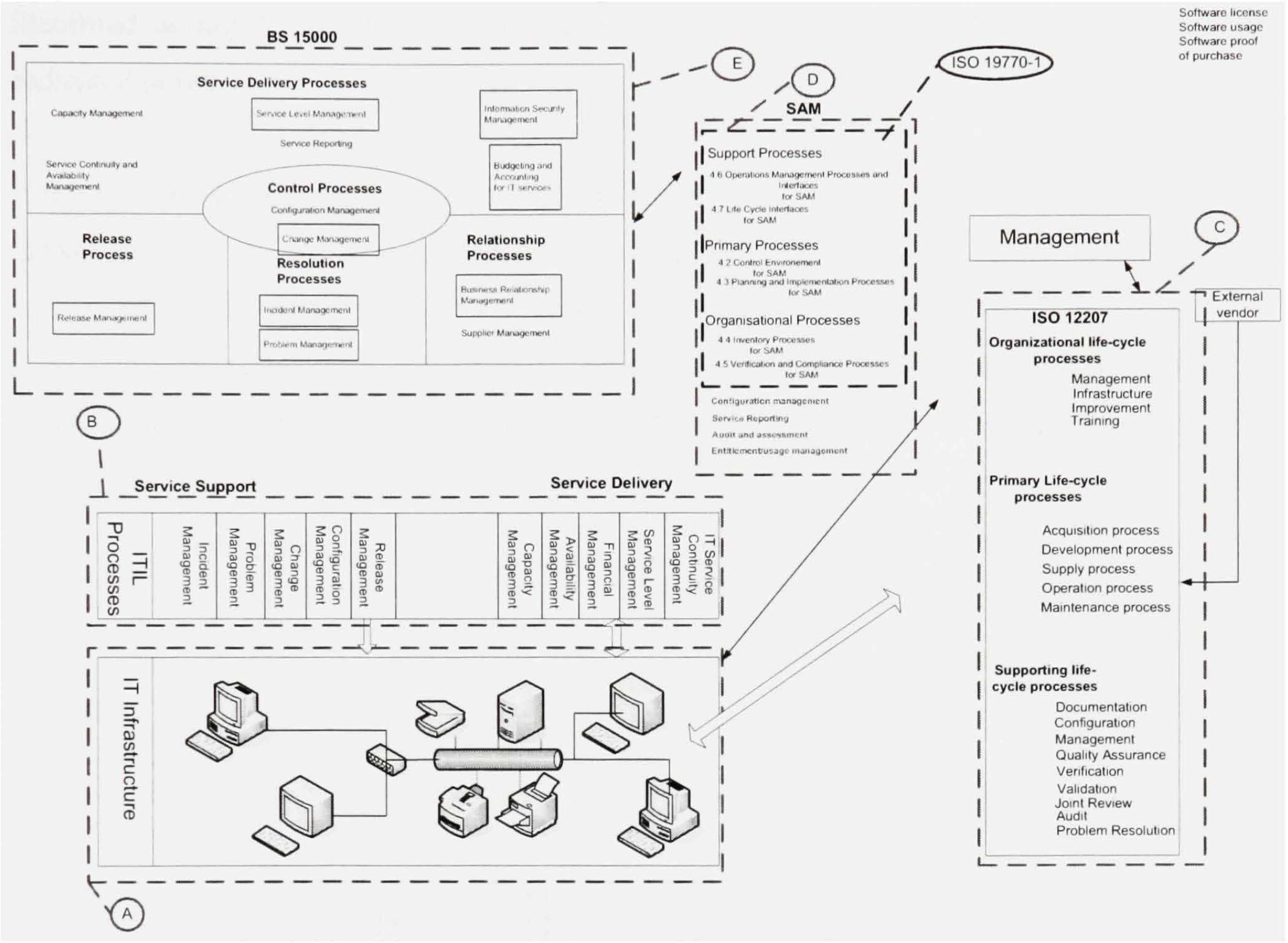


Figure 12 Positioning ISO19770-1 with BS1500 and ISO 122007

Figure 12 positions ISO/IEC 19770-1 against two other standards: BS15000 on Service Level Management and ISO/IEC 12207 on Software Lifecycle processes. Figure 12 shows that some processes from BS1500 highlighted by blue lines have been mapped unto ISO/IEC 19770-1 on SAM processes. Figure 12 also shows that ITIL is very closely linked to the IT infrastructure: each process focuses on the management of the IT infrastructure. ISO/IEC 12207, on the other hand, does not focus on the IT infrastructure: the processes of ISO/IEC 12207 apply to the entire organization. It is also important to notice that ISO/IEC 19770-1 processes do not cover every SAM processes identified during the construction on the standard; those not part of the standard are indicated in red.

If the purpose and the processes found in ISO/IEC 19770-1 were influenced by ISO/IEC 20000, the structure and grouping of processes was more influenced by ISO/IEC 12207 as detailed in CHAPTER 3 . The resulting ISO/IEC 19770-1 processes are depicted in Figure 13. The processes that have blue frames indicate processes whose purposes and labeling were influenced by those of ISO/IEC 20000.

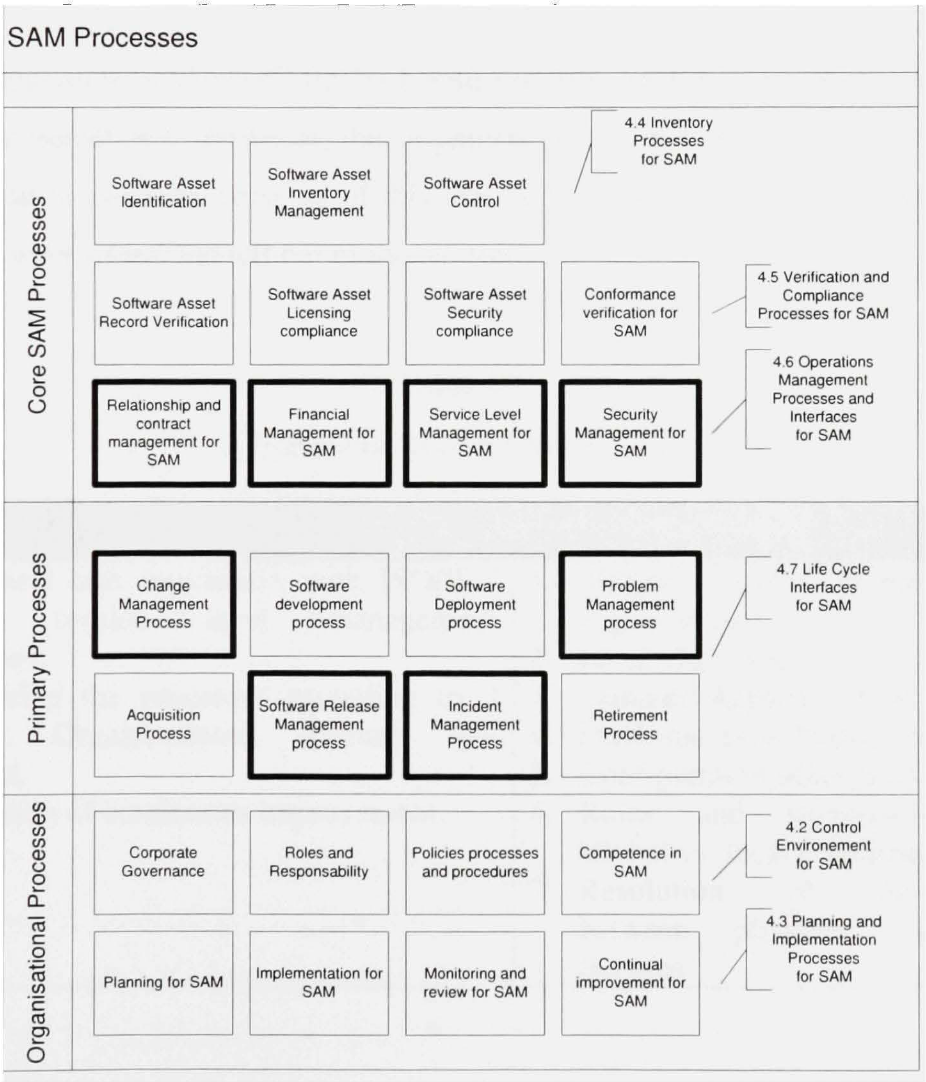

SAM Processes
Core SAM Processes
Software Asset Identification
Software Asset Inventory Management
Software Asset Control
4.4 Inventory Processes for SAM
Software Asset Record Verification
Software Asset Licensing compliance
Software Asset Security compliance
Conformance verification for SAM
4.5 Verification and Compliance Processes for SAM
Relationship and contract management for SAM
Financial Management for SAM
Service Level Management for SAM
Security Management for SAM
4.6 Operations Management Processes and Interfaces for SAM
Primary Processes
Change Management Process
Software development process
Software Deployment process
Problem Management process
4.7 Life Cycle Interfaces for SAM
Acquisition Process
Software Release Management process
Incident Management Process
Retirement Process
Organisational Processes
Corporate Governance
Roles and Responsability
Policies processes and procedures
Competence in SAM
4.2 Control Environement for SAM
Planning for SAM
Implementation for SAM
Monitoring and review for SAM
Continual improvement for SAM
4.3 Planning and Implementation Processes for SAM

Figure 13 The ISO/IEC 19770-1 standard

There is also some influence from the ITIL SAM book [19] for the development and deployment processes. Indeed, for the ITIL SAM book, "during the deployment stage, all components of a release are deployed to their agreed and approved destination environments. SAM is responsible for monitoring the deployment process so that software is only deployed to the environment for which licenses and contracts have been obtained. It is essential that during this process the details contained within the SAM database are updated to reflect the progress of the deployment project in a timely fashion. This is the stage where the most non-compliances can be introduced with regards to the use of unlicensed software within an organization."

With the alignment of ISO/IEC 19770-1 with ISO/IEC 20000, some new processes and topics were introduced. However, this alignment also forced out some other topics that were present in previous versions of this standard. Table 13 lists and summarizes the topics that were added and left out of this version.

Table 13

Variations from N3084 to 3276

New topics	Topics not retained
1. Alignment and interaction with ISO/IEC 20000 service level management processes. 2. Structuring the processes according to 3 groups: Organizational, Primary and Support. 3. The notion of continuous improvement.	1. Merger and de-merger or organization 2. Licensing complexity 3. Standard Application catalogue 4. Different Asset Management roles 5. Configuration Management 6. Roles and responsibilities of Global vs. local organizations 7. Resolution of discrepancies between physical assets and licenses

4.3.7 Official comments

In APPENDIX XV, comments from the Canadian representative outlined that continuous improvement refers to measurements but with no indication or directive on how this should be done. From the literature review and the observations of WG21, the industry is not very mature in the management of software assets; however, measuring and using measurement usually denotes a certain level of maturity: for CMMI [33, 59], organizations have to be at level 3 and up to truly benefit from measures.

4.3.8 May 2006 ISO/SC7 Plenary meeting in Bangkok, Thailand

ISO/IEC 19770-1 became an official standard in May 2006: the voting for the IS stage was accepted: this is the final stage before a standard is officially recognized and published by ISO.

4.3.9 The variations of purposes

It is important to notice how the purpose of the standard, in Table 14, has evolved throughout its development. From the start, this standard has been driven by industry and market demands: in fact it is explicitly stated in the two first versions of the standard. The variation of the purpose statement gives some indications on how the standard has evolved. For instance, the Brisbane version has the longest purpose statement which indicates an accumulation of several concepts that were not yet integrated. In 2005, with the Helsinki version, the concepts became more focused; the statement of the purpose is shorter and the 2006 version of the standard is even shorter. It is also important to notice that the two latest versions refer to the standard as being a baseline: a baseline is usually found in the configuration management process. This process is not formally present as observed by the Canadian Representative during the Montreal plenary meeting.

Table 14

Evolution of the ISO 19770-1 standard's purposes

Busan (N2622) in 2002	Montreal (N2885) in 2003	Brisbane (N3084) in 2004	Helsinki (N3276) in 2005	Bangkok - 2006
"to meet market needs/requirements to reduce costs, keeping a good software management and taking control of systems and software assets."	"to meet market needs/requirements to reduce costs, take control of systems and software assets and thus maintain status of software license compliance. It also provides the related technical guidance document that is required to support the standard"	"establishes a common framework for implementing and maintaining effective Software Asset Management Processes, in order to meet market needs/requirements to reduce costs, take control of systems and software assets and manage software asset license compliance"	"establishes a common basis for assessing whether an organization has successfully implemented to a baseline standard an integrated set of processes for software asset management (SAM)"	This part of ISO/IEC 19770 establishes a baseline for an integrated set of processes for Software Asset Management (SAM).

4.3.10 The evolution of topics and focus

Table 15 presents the topics that are covered by the four versions of the standard reviewed in this chapter. Topics that have a blue square in the "Helsinki version" column are topics that are included in the official version of the ISO/IEC 19770-1 standard for SAM processes.

Organizational processes are processes that apply to the entire organization and usually require upper management or corporate involvement. Local and global organizations are not processes per say, but are concepts important for the management of licenses in organizations that have more than one legal entity especially in more than one country where laws may vary and impact the terms and conditions of software licenses. The difference between local and global organizations has been present since the beginning and has made it into the official ISO/IEC 19770-1 SAM standard. However, the concern about managing licenses after a merger or a de-merger has not made it into the standard as it is a special case of the application of the standard. Defining the roles of the owner of the SAM processes for the global organization vs. the owner of the local processes owner did not make it into the final version of the standard; it was not a reoccurring concern and had to be dropped out of the list.

Primary processes are processes that are specific to the management of SA and their licenses. The guides and codes of conduct found in the early versions of the standard did not make it into the final version of the standard; although their validity is not in question: it was decided that the document itself should not be required; it was not important if the document was present or not, as long as the desired outcome was observed. In addition, the complexity of licenses is not directly addressed in the final version of the standard; however, the process "Competence in SAM" requires that individuals must receive the appropriate training to understand the licensing schemes of each software manufacturer that they must managed.

Most topics that were dropped are those from the group of supporting processes. This is partly due to the desire of some of the members of WG21 to have a strong alignment with ISO/IEC 20000; processes that are not present in ISO/IEC 20000 were often not retained in the final version of the standard. Amongst the processes that are not present in the final version of the standard are: configuration management, supplier management and continuity management. Both configuration management and continuity management are ITIL processes. Furthermore configuration is also present in ISO/IEC 20000: this makes it even more surprising that configuration management did not make it into ISO/IEC 19770-1. However, several configuration management activities are covered by "Inventory Processes for SAM" which includes SA identification, SA inventory management and SA control.

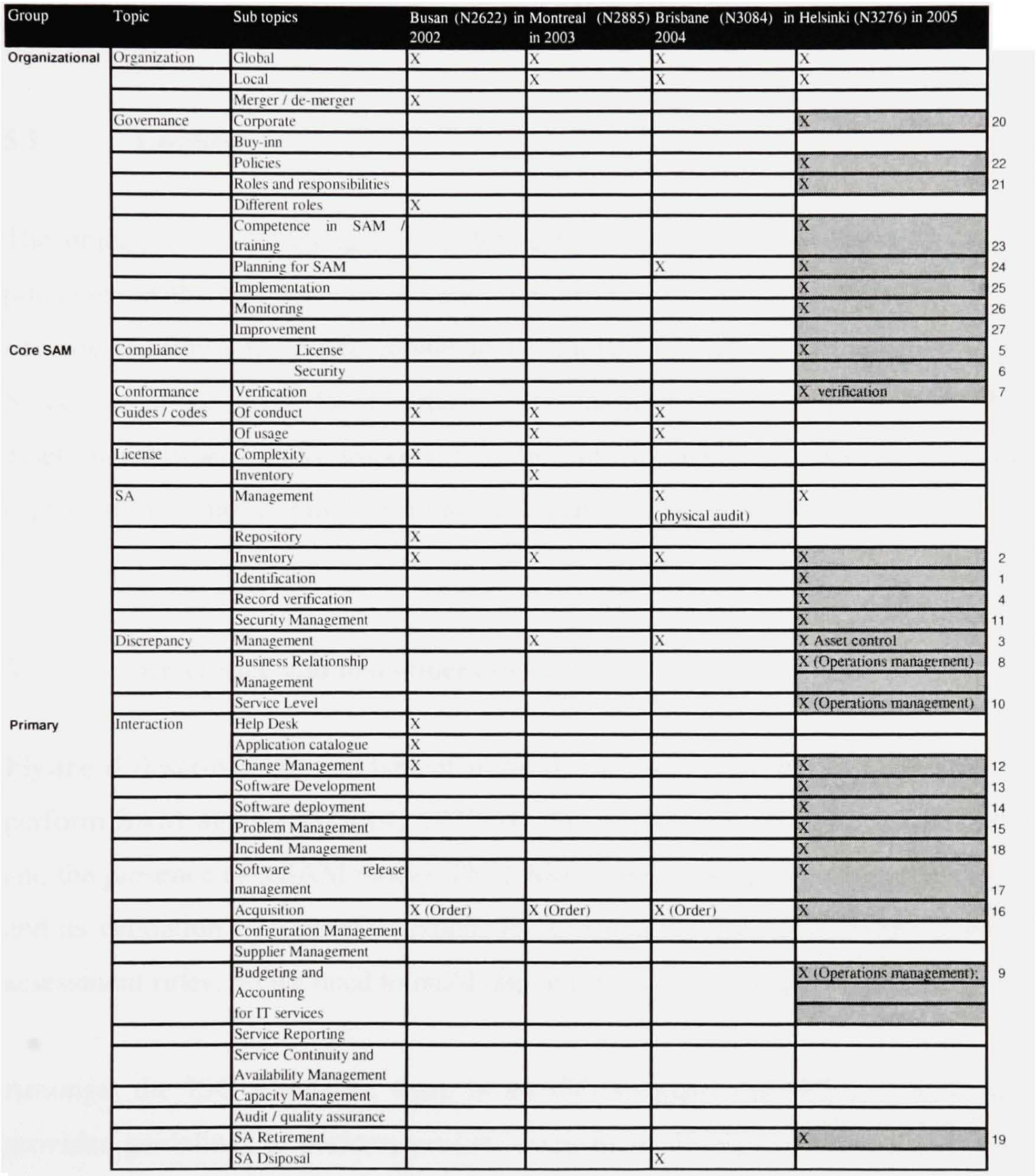
Interestingly, the last two versions of the standard used the term "baseline" in the purpose statement of the standard; despite this emphasis, only one process makes reference to a baseline which is the "SA control" that states that a baseline must be taken before performing a release.

In addition, when performing continuity management activities, it is usually required to have additional copies of software assets: software purchaser and manufacturer are both aware of this fact and a special provision is often provided by the software manufacturer to allow the software license user to perform continuity management exercises where additional copies of the software is used only for that period and for that purpose. There is no mention of that aspect of SAM in the final version of the standard.

In addition, there was a difference made between software asset disposal and SA retirement that has been lost in the final version of the standard: one was permanent (retirement) which included the removal of both the software and the corresponding license while disposal only included the removal the SA from the IT infrastructure (still available to the organization).

Table 15

Table of topics compared to the evolution of ISO/IEC 19770-1



Group	Topic	Sub topics	Busan (N2622) in 2002	Montreal (N2885) in 2003	Brisbane (N3084) in 2004	Helsinki (N3276) in 2005	
Organizational	Organization	Global	X	X	X	X	
		Local		X	X	X	
		Merger / de-merger	X				
	Governance	Corporate				X	20
		Buy-inn					
		Policies				X	22
		Roles and responsibilities				X	21
		Different roles	X				
		Competence in SAM / training				X	23
		Planning for SAM			X	X	24
		Implementation				X	25
		Monitoring				X	26
		Improvement					27
Core SAM	Compliance	License				X	5
		Security					6
	Conformance	Verification				X verification	7
	Guides / codes	Of conduct	X	X	X		
		Of usage		X	X		
	License	Complexity	X				
		Inventory		X			
	SA	Management			X (physical audit)	X	
		Repository	X				
		Inventory	X	X	X	X	2
		Identification				X	1
		Record verification				X	4
		Security Management				X	11
	Discrepancy	Management		X	X	X Asset control	3
		Business Relationship Management				X (Operations management)	8
		Service Level				X (Operations management)	10
Primary	Interaction	Help Desk	X				
		Application catalogue	X				
		Change Management	X			X	12
		Software Development				X	13
		Software deployment				X	14
		Problem Management				X	15
		Incident Management				X	18
		Software release management				X	17
		Acquisition	X (Order)	X (Order)	X (Order)	X	16
		Configuration Management					
		Supplier Management					
		Budgeting and Accounting for IT services				X (Operations management);	9
		Service Reporting					
		Service Continuity and Availability Management					
		Capacity Management					
		Audit / quality assurance					
		SA Retirement				X	19
		SA Dsiposal			X		

CHAPTER 5

SAM ASSESSMENT RULES

5.1 Context

The objective of this chapter is to define the assessment rules that will describe how processes of the reference model are going to be evaluated. This objective is important because it allows the SAM model to be applied against an organization's processes. Since one of the goals for this thesis is to contribute to the improvement of software asset management, it is important to be able to map the SAM model against the organization's current practices to identify gaps and suggest improvements.

5.2 Reference and assessment models

Figure 8 describes the assessment method used to assess an organization's ability to perform SAM against ISO/IEC 19770-1; this implies the presence of assessment rules and the presence of a SAM model. The SAM model is already described in CHAPTER 3 and its validation by a panel of experts is described in CHAPTER 4 . But what about the assessment rules, do we need to build one or can we use an existing standard?

Amongst the ISO standards, there is an ISO standard that defines requirements and provides guidelines to perform process assessment: ISO/IEC 15504. ISO/IEC 15504 is divided in several parts, one of which ISO/IEC 15504-2 [60] defines the requirements for performing process assessment for process improvement and capability determination. This portion of the standard states that process assessment is based on a two dimensional model: one containing the process dimension and another containing the capability dimension as represented in **Figure 14**. ISO/IEC 15504-2 defines the capability dimension which consists of a measurement framework comprised of six process capability levels and their associated process attributes. The reference model will be ISO/IEC 19770-1.

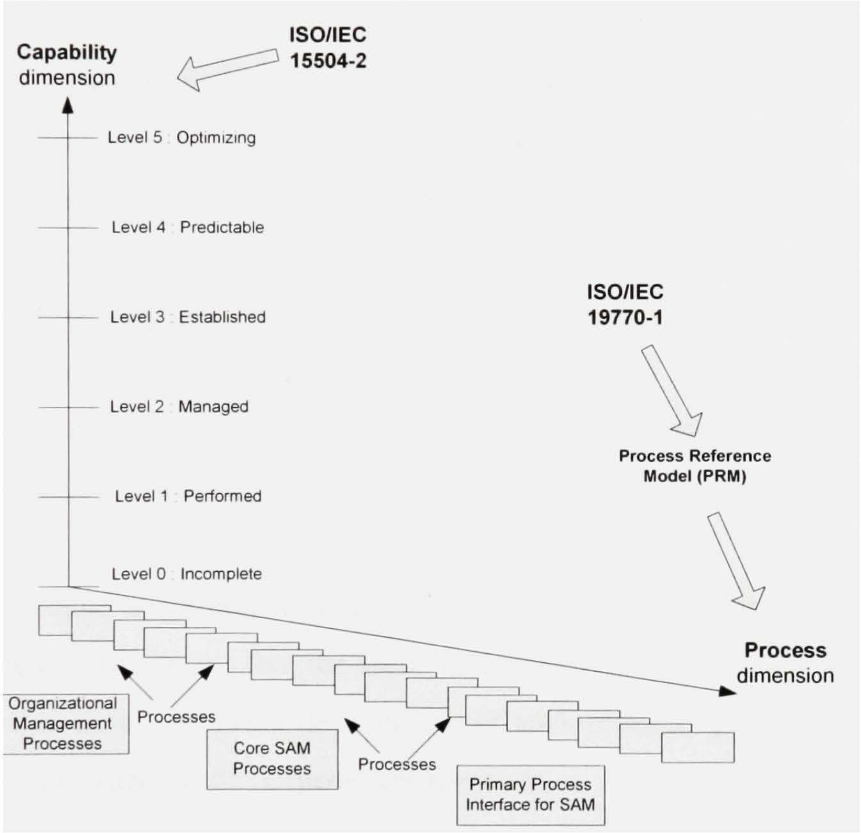


Figure 14 Process Assessment Model and its inputs – from[61]

The following sections will discuss separately these two dimensions. Since the ISO/IEC 19770-1 standard has already been described in earlier chapters, its content will not be discussed further. However, ISO/IEC 15504 states that in order for a reference model to be "assessable", the reference model must define a set of processes characterized by statements of process purpose and process outcomes.

5.3 The reference model

As indicated in **Figure 14**, the reference model used in the assessment is ISO/IEC 19770-1. Furthermore, the ISO/IEC 15504 standard specifies that the reference model must meet certain requirements:

1. It must be described in terms of purpose and outcome.
2. Reaching the set of outcomes is sufficient to achieve the purpose of the process
3. There is no measurement framework described that goes beyond level 1: this is to ensure that they are no restrictions in how the outcomes are obtained (i.e. the standard describes what is expected, not how to obtain it).

This first requirement is met since from the start, to facilitate compliance and certification of organizations with respect to this standard, all processes have clear documented outcomes that can be verified by an auditor. The purpose of each process is also clearly stated.

The second requirement states that reaching the outcomes is sufficient to achieve the purpose. This means that the focus and purpose of each process must be clearly defined and that upon observing the outcome of each process, an auditor must be able to easily conclude whether the purpose was meet or not.

Table 16 lists the processes according to the ISO/IEC 19770-1 grouping of processes and each process is described in terms of purpose and outcome (e.g. the first requirement). However, this grouping represents high level goals, if we were to stay at that level of details, the second requirement for a reference model would not be met. This is why each process grouping contains additional processes that have more detailed focused purpose. The complete list of processes is presented in APPENDIX XXII for organizational processes and APPENDIX XXIII for core SAM processes and Primary processes interfaces for SAM.

If the outcome of each of these detailed processes are met, then the purpose of each process is automatically met: this is aligned with the second requirement of the reference model according to ISO/IEC 15504.

However, the third requirement for a reference model is not met since the approach taken differs from that of ISO/IEC 15504. Indeed, ISO/IEC 15504 has taken ISO/IEC 12207 as a reference model and as depicted in **Figure 9** of CHAPTER 3 : ISO/IEC 12207 and ISO/IEC 19770-1 do not have the same level of granularity. In ISO/IEC 12207, there is no detail nor hints on how to manage the IT infrastructure: the approach in 12207 is that the industry should be free to find its own solutions. However, the industry, as indicated in the literature review, is requiring a finer level of granularity in terms of details for SAM; by design, ISO/IEC 19770-1 will not meet this third requirement.

Table 16

ISO/IEC 19770-1 assessment table

Process name	Purpose	Level 0	Level 1	Level 2	Level 3	Level 4	Level 5
Organizational processes for SAM							
4.2 Control environment for SAM	"responsibility for management of software assets is recognized at the level of the corporate board"						
4.3 Planning and implementation processes for SAM	"effective and efficient accomplishment of SAM management objectives"						
Core SAM processes							
4.4 Inventory processes for SAM	"create and maintain all stores and records for software and related assets"						
4.5 Verification and compliance processes for SAM	"detect and manage all exceptions to SAM policies, processes, and procedures"						
4.6 Operations management processes and interfaces for SAM	"execute operational management functions which are essential to achieving overall SAM objectives and benefits"						
Primary Process Interfaces for SAM							
4.7 Life cycle process interfaces for SAM	"specify SAM requirements for these life cycle processes"						

5.4 The assessment model

In **Table 16**, each process is assigned a capability rating; this rating is the output of the assessment model which itself is determined by observing specific indicators.

5.4.1 The process indicators

In 15504-5 [61], the notion of process indicators is introduced: the capability is achieved not only by demonstrating that is has achieved the described purposed but that it has the "capability" to achieve the purpose. This means that management provides guidance and that there are sufficient resources to achieve the purpose of the process. The indicators are also concerned with significant activities, resources or results associated with the achievement of the attribute purpose by a process. The indicators are grouped in three categories[61]:

1. Generic Practice (GP) indicators;
   They support the achievement of the process attribute and many of them concern management practices.
2. Generic Resource (GR) indicators;
   The availability of a resource indicates the potential to fulfill the purpose of a specific attribute.
3. Generic Work Product (GWP) indicators.
   They represent basic types of work products that may be inputs to or outputs from all types of process

However, in ISO/IEC 19770-1, organizational processes ensure that management provides guidance and communicates adequately to the organization. Furthermore, the SAM plan provides adequate funding and resource for the realization of the plan. This means that the lower level of granularity of ISO/IEC 19770-1 (compared to ISO/IEC 12207) takes away the need for the generic practice indicators and the generic resource indicators. An assessment with ISO/IEC 19770-1 will only focus on the work products of each process which are described as output of each process in APPENDIX XXII for organizational processes and APPENDIX XXIII for core SAM processes and Primary processes interfaces for SAM.

5.4.2 The rating (assessment rules)

Once the presence or absence of a work product has been observed as described in each process output, a rating must be attributed. **Table 17** provides definitions for the maturity levels listed in Table 16.

The definitions of each maturity level are taken from ISO/IEC 15504. The level 0 indicates that no work product was observed for a specific process. Level 1 process denotes that the process is not performed consistently and may vary from person to person or for the same person over time. Level 2 processes denote more consistency in the way it is performed while it is not performed consistently across the organization: if this were the case, it would be a level 3 process.

Table 17

Assessment maturity levels to be applied on SAM processes

#	Level	Definition
0	**Incomplete**	the organization fails to successfully execute the process.
1	**Performed**	process is successfully executed but may not be rigorously planned and tracked
2	**Managed**	the process is planned and tracked while it is performed; work products conform to specified standards and requirements
3	**Established**	the process is performed according to a well-defined specification that may use tailored versions of standards
4	**Predictable**	measures of process performance are collected and analyzed, leading to a quantitative understanding of process capability and an improved ability to predict performance
5	**Optimizing**	continuous process improvement against business goals is achieved through quantitative feedback

As a result of these rating rules, each process is to be rated from 0 to 5. To have an understanding of what each rating means, a more detailed analysis of ISO/IEC 19770-1 is presented in CHAPTER 6 and a graphical representation is presented in CHAPTER 7 to help identify areas that require more immediate attention.

CHAPTER 6

ASSESSMENT METHOD ANALYSIS

6.1 Context

The objective of this chapter is to provide an analysis of the SAM model to better interpret the assessment results. To this end, SAM processes are presented in a different way than in CHAPTER 3 and CHAPTER 4 to get a better understanding of how they interact with the IT infrastructure and to understand their limitations.

6.2 SAM against the IT infrastructure

In ISO/IEC 19770-1, processes are presented without context: they are grouped according to organizational, core and primary processes as depicted previously in **Figure 13**. However, this representation does not provide much help in understanding how processes interact with the IT infrastructure and more importantly where SAM related triggers and data updates must be performed.

To have a better understanding of the SAM lifecycle, **Figure 15** represents all the processes of ISO/IEC 19770-1 but represented according to where they intervene in the management of the IT infrastructure.

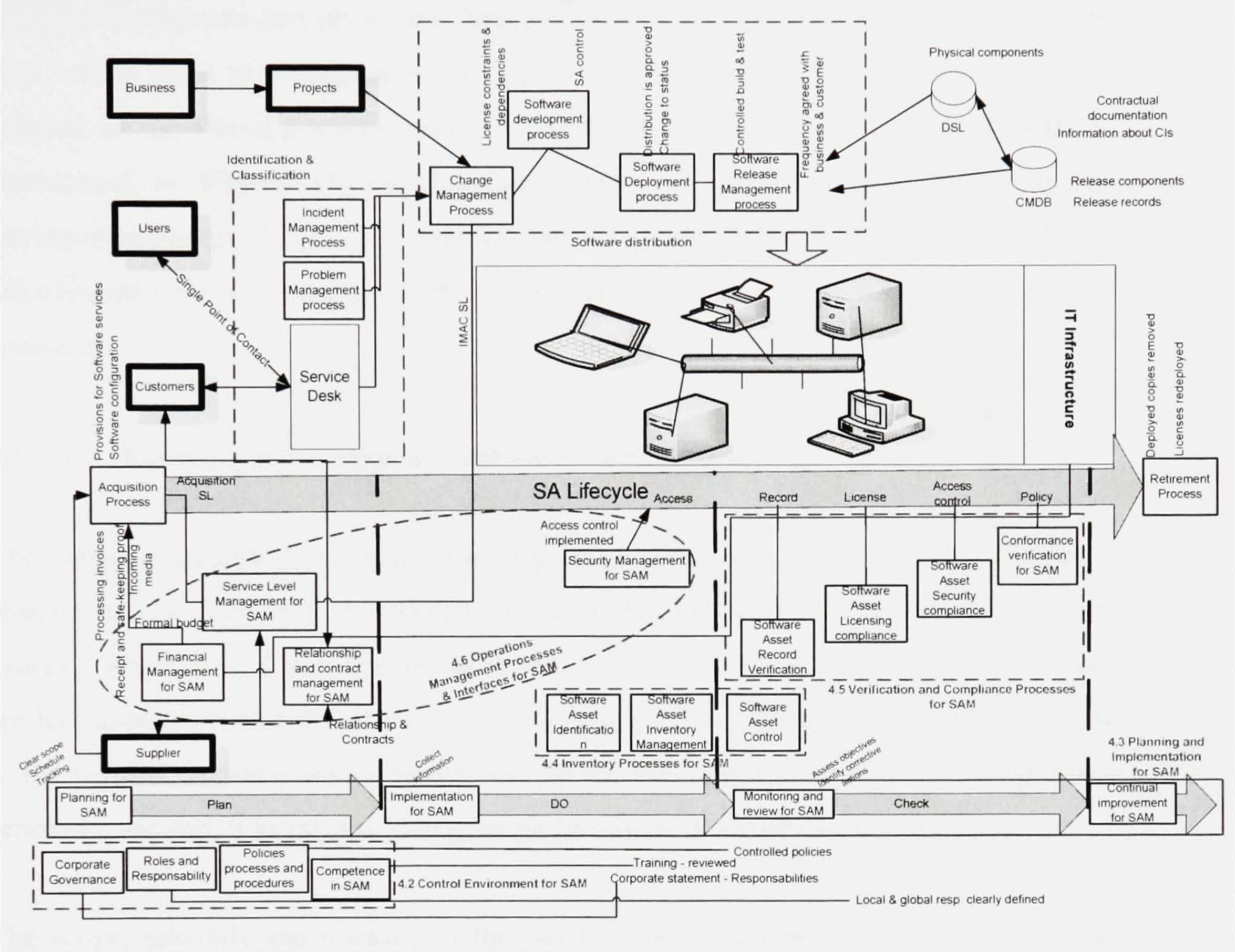

Business
Projects
License constraints & dependencies
SA control
Software development process
Distribution is approved
Change to status
Controlled build & test
Frequency agreed with business & customer
Physical components
DSL
Contractual documentation
Information about CIs
Identification & Classification
Change Management Process
Software Deployment process
Software Release Management process
Release components
CMDB
Release records
Software distribution
Users
Incident Management Process
Problem Management process
Single Point of Contact
IMAC SL
Provisions for Software services
Software configuration
Customers
Service Desk
IT Infrastructure
Deployed copies removed
Licenses redeployed
Acquisition Process
Acquisition SL
SA Lifecycle
Access
Record
License
Access control
Policy
Retirement Process
Access control implemented
Security Management for SAM
Conformance verification for SAM
Software Asset Security compliance
Software Asset Licensing compliance
Software Asset Record Verification
Processing invoices
Receipt and safe-keeping proof
Incoming media
Service Level Management for SAM
Formal budget
Financial Management for SAM
Relationship and contract management for SAM
4.6 Operations Management Processes & Interfaces for SAM
Software Asset Identification
Software Asset Inventory Management
Software Asset Control
4.5 Verification and Compliance Processes for SAM
Supplier
Relationship & Contracts
4.4 Inventory Processes for SAM
Clear scope
Schedule
Tracking
Collect information
Assess objectives
Identify corrective actions
4.3 Planning and Implementation for SAM
Planning for SAM
Plan
Implementation for SAM
DO
Monitoring and review for SAM
Check
Continual improvement for SAM
Corporate Governance
Roles and Responsability
Policies processes and procedures
Competence in SAM
4.2 Control Environment for SAM
Controlled policies
Training - reviewed
Corporate statement - Responsabilities
Local & global resp clearly defined

Figure 15 ISO19770-1 against the IT infrastructure

6.2.1 SAM management processes

At the very bottom of **Figure 15** is the management or organizational processes. These processes do not directly interact with the infrastructure but ensure buy-inn from the organization, sustain commitment and formal allocation of resources and funding. These processes are grouped under "control environment for SAM" and are oriented towards upper management and produce policies, directives, budgets and guidelines for other processes.

The set of processes just above are depicted by four arrows labeled "Plan, Do, Check, Act" which refer to management principles first popularized by Deming (sometime referred as the Deming wheel) and are borrowed from ISO 9001 and BS15000. As represented in **Figure 15**, the four processes are grouped under "planning and implementation for SAM" and interact more with other processes than with the IT infrastructure itself. These four phases are discussed further in the following four sections.

6.2.1.1 Planning and acquiring Software Assets (Plan)

The budget allocated from financial management allows the purchase of goods from a selected set of suppliers determined by the Relationship and Contract management process. The Service Level management process manages the acquisition service levels: the lead time to receive the software is one example. The software will need to conform to the security access controls determined by Security management for SAM. All these processes are part "Operations Management Processes & Interfaces for SAM".

The scope, schedule and tracking of the purchase are first determined by a SAM plan which is located at the bottom right of **Figure 15** just at the beginning of the gray arrow that says "Plan".

6.2.1.2 Implementing SAM (Do)

Before being able to monitor the implementation of SAM, information must be collected according to the scope described in the planning process. The scope of the information to be collected is determined by the "implementation for SAM" process. When interventions are made to the infrastructure such as the installation and removal of software, the information described in the SAM plan has to be collected.

The top left portion of Figure 15 presents the processes that receive requests to Install, Move, Add or Change (IMAC) software assets; there are two processes (incident and problem management) and one service (the service desk). In ISO/IEC 20000, incident and problem management are referred to as resolution processes because they are usually used to restore the IT infrastructure to a "stable" state or a working state. The service Desk is present only in ITIL: it is the single point of contact that users and customers use to make requests such as purchases and repairs. These processes do not install or remove the software: they only generate the requests to do so.

The installation or modification of software is managed by the set of processes at the very top of Figure 15 with the red dotted line with the label "Software distribution". The first of these processes is the "change management process" which determines if and when the change will take place after an evaluation of the impact of the changes has been made. The other three processes all play a key role in ensuring that the software does not impact negatively the IT infrastructure and that the terms and conditions of the license are respected. The "Software development process" ensures that the license constraints and dependencies are identified and considered as well as how its access controls will be monitored. The "Software Deployment process" ensures that the distribution of the software is approved and that any change to its status (such as development environment vs. production environment) is recorded. Finally, the software is physically introduced into the IT infrastructure via the "Software Release management process" which controls the builds and tests of the software as well as agreeing with the business and customer on when these distributions will happen. At the end of this process, the software is installed or modified if it meets all the requirements of the organization. At each stage, key information must be captured and recorded; labels besides each process are a subset of this information.

To make sure that the plan is followed and that no unwanted software is introduced, the IT infrastructure is monitored. During the planning process, the "Implementation for

SAM" process determines what information is necessary to collect in order to ensure that the plan is followed. The collection of the information in the context of SAM is represented by the inventory processes for SAM indicated by a red-dotted line above the grey arrow marked as "Do". The "Software Asset Identification" process identifies all physical occurrences of software while the "Software Asset Inventory Management" ensures that they are data stores and policies to support all reconciliation processes of the data collected in the previous process. The reconciliation process is managed by the "Software Asset Control" process which ensures that all changes are recorded and approved.

6.2.1.3 Monitoring the Software Assets (Check)

To ensure that the process of managing the Software Assets is working properly, the verification process looks at various aspects of the SAM process. The "Software Asset Record verification" process ensures that the information that is collected is appropriately recorded. The "Software Asset Licensing compliance" process ensures that the organization owns the software and that a reconciliation process is in place when discrepancies are observed. The "software Asset security compliance" process verifies that access control requirements are respected in the installation and usage of the software and the "Conformance verification for SAM" ensures that organizational policies are applied appropriately. These four processes are labeled as "Verification and Compliance Processes for SAM" and are above the "Check" gray arrow at the bottom left of Figure 15. The amount of effort and resources dedicated to these processes should be aligned with the objectives defined in the "Monitoring and review for SAM" process.

6.2.1.4 Continual improvement (Act)

Although there are some operational reconciliation processes such as Software Asset Controls, it is the continual improvement for SAM that handles suggestions and improvement initiatives to change the SAM process. This process will ensure that suggestions for improvements are captured and are considered for the next planning initiative for SAM: if the planning is done yearly, the suggestions captured during the previous year are used as input for the plan of the following year. However, before being accepted into the new SAM plan, the impact of implementing the suggestions must be evaluated: this requires to go through the IT change management process in place in the organization.

6.3 Configuration management

6.3.1 Rationale

The purpose of configuration management (as per ITL and BS15000) is to account for all the software within an organization not only at the moment of purchase but throughout its entire lifecycle: software evolves and changes when patches, fixes and upgrades are applied to the initial software. In addition, the license that was originally linked to the purchased software, may also change as manufacturers adapt their licensing schemes to maximize their profits. To control and match changes brought to both software and license, a snapshot or a baseline has to be taken when the product is purchased so that changes can be tracked latter on.

Configuration management provides the information and the relationships necessary for the other processes to manage the infrastructure; here a special attention is brought to the management of software assets and the terms and conditions of the software license.

6.3.2 Configuration and SAM

All the processes presented in **Figure 15** assume that the appropriate information is available to assess the situation and determine the appropriate corrective action. But this information must be managed and structured in order to meet the needs of each individual process. In ITIL and ISO/IEC 20000, Configuration Management is responsible for ensuring this information is available. However, in the latter two references, the concern and purpose of Configuration Management is very operational: ensure the stability of the IT infrastructure and making the software work. That is to say that contractual information such as support contract and constraints and restrictions of licenses are not typically found in the configuration management data base CMDB.

Since both ITIL and ISO/IEC 20000 have configuration management as part of their processes, it is somewhat surprising that ISO/IEC 19770-1 does not. It does have however software asset controls which ensure that SAM related information is recorded.

6.3.3 Enhancing the Configuration Management Database (CMDB)

There is a viewpoint that the CMDB could hold this additional information about software assets: but the definition of the CMDB in ITIL typically does not. In ITIL it simply requires that components of the infrastructure (i.e. software, hardware) are stored in the CMDB along with their relationships. The rationale is that if a component changes, it must be possible to easily assess which other components are affected to assess its impact. Furthermore, ITIL defines a DSL (Definitive software library) as a placeholder where all legitimate software copies are stored. Both the DSL and the CMDB must be updated by the processes listed in Figure 15 as required.

6.3.4 Process triggers to the CMDB

The CMBD will be updated when software are installed, moved, changed or removed. The CMDB may also change when a corrective action is implemented to return to a compliant state. The CMDB may also be updated at purchasing time to update financial, contractual or manufacturer information. In the literature review, as illustrated in **Figure 2**, there was no defined control mechanism for the organization to manage and control software assets within the IT infrastructure. With the release of ISO/IEC 19770-1, as illustrated in **Figure 16**, the organization has now some control over the software assets being introduced into the IT infrastructure.

The level of control over Software assets will be mostly determined by the level of maturity of each processes of ISO/IEC 19770-1 which, in turn, will determine how the data collected is accurate and adequate for the needs of the organization.

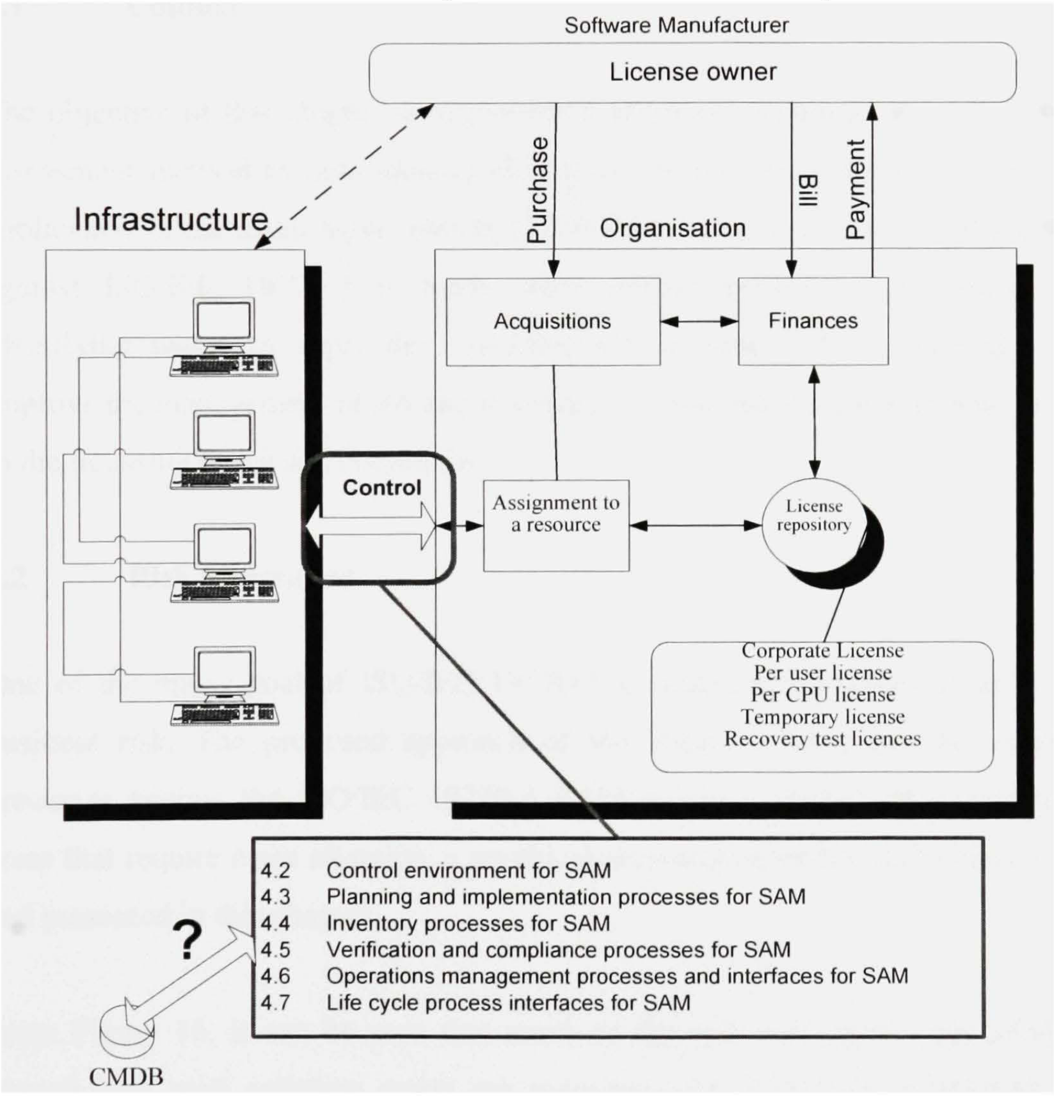

Software Manufacturer
License owner
Purchase
Bill
Payment
Infrastructure
Organisation
Acquisitions
Finances
Control
Assignment to a resource
License repository
Corporate License
Per user license
Per CPU license
Temporary license
Recovery test licences
4.2 Control environment for SAM
4.3 Planning and implementation processes for SAM
4.4 Inventory processes for SAM
4.5 Verification and compliance processes for SAM
4.6 Operations management processes and interfaces for SAM
4.7 Life cycle process interfaces for SAM
?
CMDB

Figure 16 ISO 19770-1 and the control of the IT infrastructure

CHAPTER 7

PRESENTATION OF THE RESULT OF AN ASSESSMENT

7.1 Context

The objective of this chapter is to provide a graphical presentation of the result of the assessment method to help identify risk areas for the organization; as a result of the application of the assessment rules in CHAPTER 5 , an assessment of the organization against ISO/IEC 19770-1 is made. This chapter provides a graphical means of identifying where to apply the corrective actions that will be required in order to improve the management of SA and to reduce the risks associated with non conformance to the licensing terms and conditions.

7.2 Risk assessment

One of the major goal of ISO/IEC 19770-1 as stated in its introduction is to reduce business risk. The proposed approach of this thesis is to assess the organization's processes against the ISO/IEC 19770-1 SAM reference model. However, to identify areas that require more attention, a graphical presentation of the assessment is proposed and presented in this chapter.

From Figure 16, it can be seen that much of the risk associated with SAM revolves around how well software assets are managed and controlled. If ISO/IEC 19770-1 defines processes to manage and control software assets; the level of maturity of each process should give a good indication of the level of risk the organization is facing.

The assessment is performed by applying the assessment rating rules described in CHAPTER 5 . **Table 16** contains the results of the assessment presented by the groups of processes found in ISO/IEC 19770-1. To obtain more details on the result of the assessment, the result for all the assessments can be view in **Table 29** and **Table 30** found in APPENDIX XXII and APPENDIX XXIII would provide a detailed view of the resulting assessment.

Figure 17 is the graphical presentation of **Table 17**; the graphical presentation allows to identifying the areas that are more at risk. The graphical presentation adds even more value when there is a great number of a processes; ISO/IEC 19770-1 has 27 processes. This is why presenting graphically both **Table 29** and **Table 30** can be beneficial: if not, each table fills several pages and identifying where to focus requires more time. On the other hand, **Figure 18** allows to identify the weak processes on a single diagram as opposed to several pages as in APPENDIX XXII and APPENDIX XXIII.

In this illustrative example, the Kiviat diagram of **Figure 17** indicates that Life cycle processes for SAM are mature but that, however, inventory processes for SAM are not; this is where any corrective or improvement initiative should start.

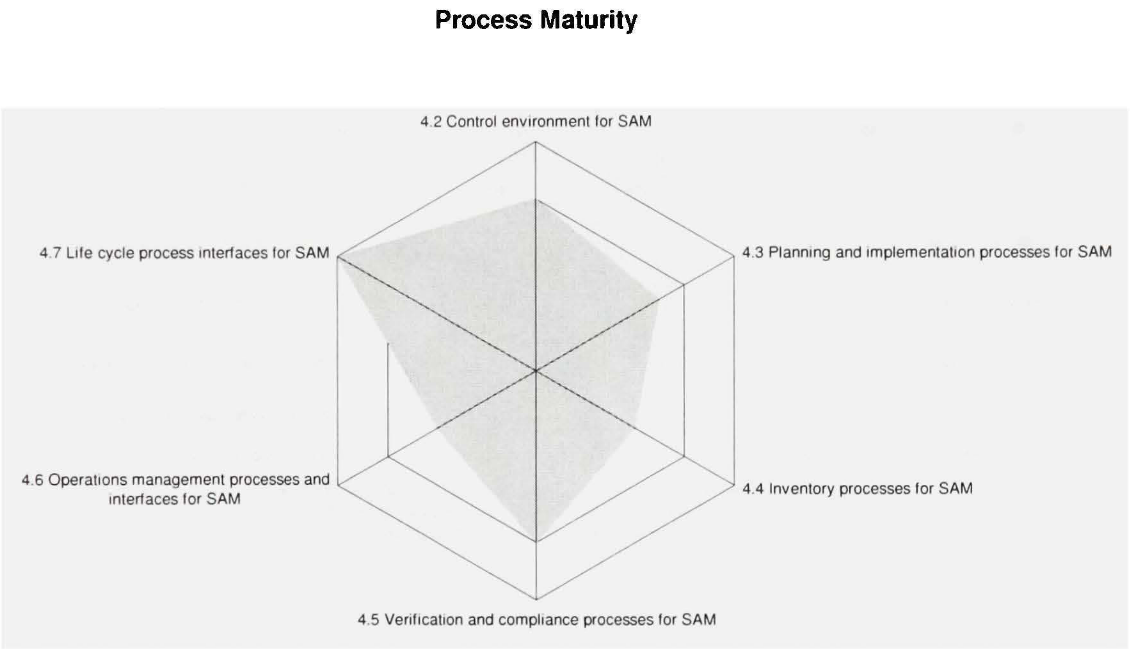


Figure 17 Risk assessment through the maturity profile of an organization

The ISO/IEC 19770-1 standard on SAM processes has two levels of processes. The first level is represented by **Figure 17**. To have a better understanding of how well the organization is performing against the SAM reference model, the results of the assessment against the second level processes have to be studied is represented by **Figure 18**.

Figure 18 provides more insights on the weaknesses and strengths of the organization in terms of SAM. In this illustrative example in Figure 22, service level management should be prioritized in order to reduce the risk associated with not managing well SA.

Detailed graph

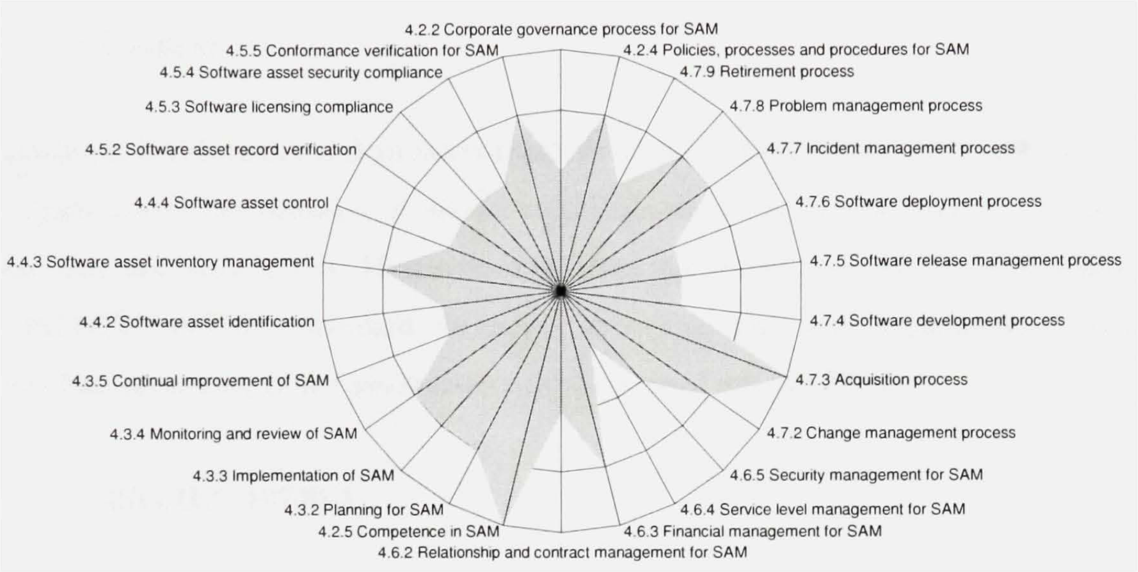


Figure 18 Detailed presentation of the assessment results

CHAPTER 8

INDUSTRIAL IMPACT ON STANDARDIZATION

8.1 Context

Proposing a SAM model and an assessment method can have lead to several benefits to an organization; for instance, it allows organizations to assess themselves and detect where they are more at risk. However, it requires that the industry progressively adopts the ISO/IEC 19770-1 standard on SAM processes and that organizations assess themselves according to the assessment rules described in CHAPTER 5 .

8.2 ISO/IEC 19770-1

8.2.1 Adoption of ISO/IEC 19770-1 by the industry

The ISO/IEC 19770-1 standard on SAM processes has been published and made available to the industry in May 2006. Penetration of any ISO standard in the industry is typically spread over many years and will vary considerably: some standards provide guidance only while others become, over time, mandatory tools of trade. In both cases, the ISO organization does not monitor standards adoption rates.

The adoption of the ISO/ 19770-1 standard has already generated interest in the industry, as illustrated by the presentations and interviews been given by the author of this thesis. In particular, interviews were published in the March 2006 edition of the RedMond Channel Partner Magazine[62] and a presentation accepted in May 2006 at the SAM Summit 2006 in Chicago [63]; this presentation was as well distributed in a WebCast to Microsoft one month latter.

Through such conferences from WG21 members, including the author of this thesis, the industry is becoming more and more aware of the standard; web search trough Google suggests a growing number of consulting organizations referring to the standard. Only time will tell if organizations adopts it or not.

8.2.2 Assessment vs. compliance

From the start, ISO/IEC 19770-1 has been designed with the intent of being used to verify compliance against the standard; a checklist that would require compliance to the entire standard without levels of maturity. This has greatly influenced how the outcome of each process has been described. However, this is only the first step. In order to recognize compliance to a standard, a certification scheme must be developed and a certifying body must accept to own the certification scheme. At the timing of submitting this thesis of the writing, the UK, through UKAS (United Kingdom Accreditation Service) is running two competing accreditation schemes in a pilot mode; the UKAS requires that a pilot on the accreditation scheme be ran successful for 18 months in the industry before accepting the scheme.

My personal feedback from the industry suggests that the industry is not ready to go into another certification scheme unless they are not forced into it. This is why the assessment method is recommended: it would identify the weak areas of SAM in organizations and provide improvement suggestions.

8.2.3 Risk assessment model

The application of the assessment model on ISO/IEC 19770-1 can help organizations better cope with the risks associated with the illegal use of software assets. By identifying where the organization is loosing control over the unauthorized used of software assets, organizations can know where to take corrective actions.

Furthermore, the graphic representation of **Figure 18**, provides a global view of the result of the process. The use of such an assessment model with its graphical representation can lead organizations to acquire additional knowledge and expertise in managing software asset by recognizing potential risks associated with a specific graphical profile. In addition, over time, organizations will also learn to monitor and associate specific corrective actions simply by observing how the SAM graphical profile (i.e. **Figure 18**) looks and changes over time. For example, if corrective actions are taken in the planning phase, the evolution of the graphical representation can confirm which processes are affected and how long it takes for the changes to be observed.

8.3 The SAM TAG (ISO/IEC 19770-2)

Currently the identification of software and their reconciliation to their commercial name is a time consuming process as described in the literature review. ISO/SC7 is working, along with the industry, towards a tag that would allow for a much more systematic identification and reconciliation process of software with their commercial name. The definition of what is SAM (ISO/IEC 19770-1) also helps this initiative since it is now possible to specify where in the software asset lifecycle, this inventory and reconciliation takes place. It also allows the industry to adopt a common vocabulary to describe what is SAM.

The industry participation and interest is growing, as attested by the development of the next standard ISO/IEC 19770-2 on the SAM TAG. As described in the literature review and in the motivation for the development of the ISO/IEC 19770-1, the identification of software discovered is an important concern for the industry. This proposed ISO/IEC 19770-2 is currently scheduled to be published in 2009. But the relevance of this upcoming standard is already getting promoted. For instance, I made a presentation in June 2006 to the British Standard Institute [64] concerning the future SAM TAG and its intended scope. Furthermore, an agreement was established between WG21 and the IBSMA: International Business Software Manager Association (http://www.ecpmedia.com/ibsma_samstds.html) to conduct industry consultations with software manufacturers and software purchasers. The IBSMA has partnerships with Microsoft, IBM, Adobe and many important software purchasers in the United States

These discussions with the industry have also brought out some concerns about defining what is software entitlement: for example, how to determine without ambiguity if the organization has the right to install and use a specific software? Presently the terms and conditions are not clear enough and require further discussion and sometimes negotiations between the software buyers and the software manufacturers. If discussions on the topic can lead to clear terms and conditions that would also have an important impact on the industry.

CONCLUSION

Motivation for this thesis

An important motivation for this thesis was to understand why the industry had so much trouble with Software Asset Management (SAM). To improve on the current situation, this thesis proposes and describes a common set of processes to define what is SAM and proposes a Kiviat representation of the result of a SAM assessment to help organizations identify SAM related risks. This definition of SAM provides the industry with a common understanding of what is in the scope of SAM and a common set of vocabulary to describe the content of SAM.

State of the art and practice on SAM at the beginning of this thesis

At the beginning of the writing of this thesis, SAM management was not clearly defined and the perceptions of what was SAM varied according to the sources of references:

- For tool vendors: SAM was considered as a suite of tools that could be sold but for which the functionalities varied from vendor to vendor with no agreement on the vocabulary to describe the scope of those functionalities.
- White papers and industry consulting firms such as Gartner were quoting ITIL as a good start but as an incomplete source of reference. Furthermore, to define and manage better SAM, the industry did not agree on alternatives to ITIL.
- Book on software assets management (SAM) were almost non existent only the OGC group, the owner of ITIL, had proposed a book specifically on the subject but the industry had not adapted it as an industry solution for SAM.

- Software engineering had very little literature on SAM; however other engineering disciplines had developed tools, techniques and processes about inventory and asset management; all of these concepts, however, had been mostly applied to physical assets only.

The next paragraphs present a summary of the results achieved through this thesis

Objective 1: Actively contribute to the development and to the content of the ISO international standard on SAM (ISO/IEC 19770-1)

This research work was carried out concurrently to the ISO work on software asset management. The ISO working group initially included only six practitioners from industry, and the group size increased progressively. Expertise of practitioners in ISO group is typically based on individual expertise acquired in industry and is often context dependent.

The author of this thesis joined the ISO working group (ISO/IEC JTC1/SC7 WG21) at the beginning of both his thesis and concurrent initiation of the ISO work on SAM, and has since been an active contributor, including participation at semi-annual international meetings of the working group. The literature review within, and outside of, the software engineering field, has helped identify various components to the model finally adopted in the ISO international standard on SAM. A number of SAM components identified however did not get to be included within the ISO standard, but were still documented for later use by the international standardization community ; the additional components were documented in Chapter 4.

The contribution by the author of the thesis were especially influential to the standard before 2005 when the core group of experts for WG21 present at the bi-yearly meetings rarely exceeded 6 members. The contribution by the author remained important throughout the development of the standard, but less influential as the group grew in size and popularity with industry. In summary: the first objective of contributing actively to the content of the international standard was met and led to the publication of ISO/IEC 19770-1 in May 2006.

Objective 2: Capture, identify and analyze elements that are relevant to SAM, including those that would not make it into the final version of the international standard.

During the development of the international standard on SAM, several concepts and elements did not make it into the standard: the fact that these elements, temporarily discarded, were brought up by experts in the field meant that they were important concerns for at least some of the industry participants.

For a better understanding of the current scope of SAM, it is important to understand as well the elements that have not yet included in the international standard: such elements, even though being currently still debatable, are, however, of interest to a number of experts in this field. Typically, all these elements of discussions are not included in the official published of May 2006 standard.

For instance, the complexity of licenses is not a topic included within the published ISO/IEC 19770-1 standard. However, it remains an important concern for the industry and this may affect the level of training required by the SAM manager and all others that need to maintain and manage those licenses; such elements not part of the published standard, have been documented in this thesis. Chapter 4 has therefore identified and analyzed elements that are relevant to SAM, including those that did not make it into the final version of the international standard.

Objective 3: Provide an analysis of the international SAM standard with respect to the 27 processes within ISO/IEC 19770-1

In order for the industry to understand and apply ISO/IEC 19770-1, it is important to provide additional context and explanation on how the 27 processes relate to an organization. Chapter 4 (through Figure 12), positions ISO/IEC 19770-1 against other existing standards while outlining similarities and differences. By examining similarities and differences the delineation of scope and perspective (i.e. granularity) becomes clearer which also helps to understand why some elements did not make the final version of the standard (i.e. for instance some elements are already mentioned in other standards).

The ISO/IEC 19770-1 standard on SAM processes provides a list of 27 processes but does not indicate how they relate to the organization's IT infrastructure and how they relate to each other. This is why Figure 15 and its descriptive text provide additional insights not provided on the SAM international standard: each process interacts with the IT infrastructure and/or other processes: without this insight applying the SAM standard may vary greatly according to the level of SAM and ISO/IEC 19770-1 expertise. This is an undesirable effect since the SAM standard aims to obtain a more standardized SAM environment and set of practices. Chapter 4 and Chapter 6 provides important information that allows better analysis of the international SAM standard with respect to the 27 processes within ISO/IEC 19770-1

Objective 4: Develop an exploratory assessment method to allow organizations to determine their gaps against ISO/IEC 19770-1.

When organizations whishes to implement some organizational changes, they must determine what goals they want to reach, where they are today and how to fill the gap between their current situation and their goals. Although ISO/IEC 19770-1 can be used

to define the SAM goal, it does not describe how to determine gaps between the organization and the international SAM standard.
To determine how wide is the gap between their current situation and their desired goals organizations must perform an assessment of where they are compared to their goals (i.e. ISO/IEC 19770-1).

To perform an assessment, the first step is to determine what rules will be used to rate the organization against the SAM processes of the international standard. Chapter 5 provides such a rating mechanism by using the assessments concepts of ISO/IEC 15504. However, ISO 15504 has some requirements that ISO/IEC 19770-1 does not meet (i.e. such as all processes described as level 1 process).

To get around the assessment restrictions of ISO/IEC 15504, a more thorough analysis of ISO/IEC 19770-1 is performed in Chapter 6 to provide more background information to allow interpretation of the assessment results. Indeed, Chapter 6, describes the relationships between the 27 SAM processes and between the IT infrastructure.

To summarize all this information, the contextual information of Chapter 6 is combined with a graphical representation to simplify its application in the industry. The result is a Kiviat diagram with a set of guidelines that can be used to assess an organization's gap against ISO/IEC 19770-1 and prioritize the most urgent SAM risks. Using the information provided by the details analysis of chapter 6 with the summarize view of chapter 7, organization can assess their current situation and rate the gap against ISO/IEC 19770-1.

Future work

The thesis does not contain yet any empirical analysis of the maturity model proposed. This section identifies some limits and constraints that still must be addressed:

- Once the industry will have adopted the ISO/IEC 19770-1 standard for a longer period of time, it will be possible to validate if the assessment of the gaps against ISO/IEC 19770-1 is a viable mean of identifying weak points. If data can be consistently captured and compiled, statistical method can be used to perform analysis and validations. Furthermore, the study of the evolution of SAM through the experimental assessment method can also be used to better estimate and even measure SAM related risk in the organization: presently organizations have no means to quantifying the risk associated with poor SAM.

Much of the difficulties with managing SAM come from the fact that the industry is trying to manage terms and conditions of licenses which are defined in legal terms. A goal is to be able to know immediately, and in an automated fashion, if a software can be installed or used: some researchers in the industry are referring to this as "entitlement":

- Terms and conditions found in software licenses are sometimes vague and subject to interpretation; the industry needs to better define these terms and conditions if some automation is to be made possible in terms of monitoring and control while meeting the users' needs (i.e. not just simply buying more licenses in the fear of breaching terms and conditions).
- IT operations are mostly concerned with maintaining the IT operations in a working state to meet predefined levels of service. The difficulty of interpreting the terms and conditions found in licenses means that IT operations do not pay much attention to them: this puts organizations at risk. Since these restrictions represent mandatory legal obligations, but often without physical mechanisms to enforce the terms and conditions, it is up to operational personal with important workload to interpret complex restrictions sometime based on usage rules and expressed in legal terms.
- The definition of scope of SAM and the standardization of terminology provided by the ISO/IEC 19770-1 international standard on SAM allows to

> better define control points to monitor how well an organization manages SAM. But the completeness of those control points cannot be tested until industry data becomes available. This will not only require data from the industry but also empirical feedback on how well the international standard helps organizations and where it fails to do so.

This means there is still much work to be done in order to better define terms and conditions of licenses to allow the introduction of defined control points and ultimately automation of the verification and validation of licenses compliance. It also means that empirical data will be key in determining what research issues are the most important for the industry.

APPENDIX I

MARKET WHITE PAPERS

The following is a list of papers by industry analysts. Since these are not in the public domain, only a short description is presented.

Title	Description
Asset Management in a CSD: Square Peg in a Round Hole?[6]	The consolidated service desk, as the single point of contact, requires that agents capture information about incidents, problems and trigger processes that can generate changes to assets: the existing configuration management does not need asset management but these changes can affect it. Tool vendors do not provide a solution: processes need to be clarified before an automated solution can be effective.
Gaining Efficiencies With PC Life Cycle Management[7]	Having a life cycle strategy often brings clarity and ease enterprise communication; it should include an enterprise procurement standard for hardware and include and a physical management policy to analyze the physical makeup of the installed base
IT Asset Auto-Discovery and Inventory Tool Q&A[1]	It is not sufficient to use auto-discovery tools, to manage its assets, an enterprise must know what assets it owns, where it is allocated, how it is used and how it is changing over time; having a snapshot without this history does not offer that information.
IT Asset Management: Closed Loop or Flat Line?[10]	Asset management is a continuous process; part of its success is attributed to recognizing the link between phases of the procurement process.
IT Assets, Inventory and Configuration: How Do They Differ?[4]	Vendors refer to IT asset management, inventory and configuration management interchangeably, yet these technology areas are different; the overlap is purely in the data and only be clarily defining the enterprise's needs can the enterprise select the appropriate tool/process to start with.
ITIL Process Gaps for Asset and Configuration Data[5]	Gartner defines asset and configuration management differently than ITIL : Gartner's configuration management definition is broader, encompassing software distribution, along with other components, such as remote control and license metering.
PC Disposal: Data Security and Sanitization Responsibilities[8]	When disposing of surplus or obsolete PCs , it is important to specify that data must be sanitized and specify other security procedures.
PC Disposal: Methods for Secure Data Sanitization[9]	Enterprises must sanitize and not simply clean when disposing hardware with data**:** "Clearing" does not prevent the recovery of data, sanitization does. Some techniques are

Title	Description
	disucsses such as *Degaussing, Overwriting and Total destruction.*
Understanding the Key Attributes of IT Asset Management[2]	"There is a lot of confusion in the marketplace regarding the term "asset management," especially from an IT context; vendors have added to the confusion by using asset management to describe auto-discovery and inventory functionality, rather than the complete functionality of an IT asset management repository."
Workflow Processes for Tracking and Buying Your IT Assets[3]	"IT asset management is only as effective as the processes that support it, enterprises have longed for ways to automate these processes. The introduction of e-procurement modules and workflow engines provides significant progress toward resolving customer dilemmas."
Common Mistakes in IT Asset Management[12]	It is about the process not the tool, Auto-discovery agents cannot discovery everything; some processes will be manual, evaluate your needs before purchasing a tool, don't start too complicated, document do not rely only on a project champion.
The Total Economic Impact™ of Integrated IT Asset Management and Help Desk[13]	Discusses when and what types of organization benefit the most from integrating help desk with asset management (size of the enterprise and product integration are important factors). Note: Asset Management and Configuration are not clearly differentiated in this paper.
Configuration Management: An IT Operational Foundation[11]	Discusses how organization will move beyond simple desktop, server, and network configurations, explains ITIL and positions Inventory Management (tracking) vs. Asset Management (Financial).

APPENDIX II

SAM TOOL VENDORS

List of software vendors:

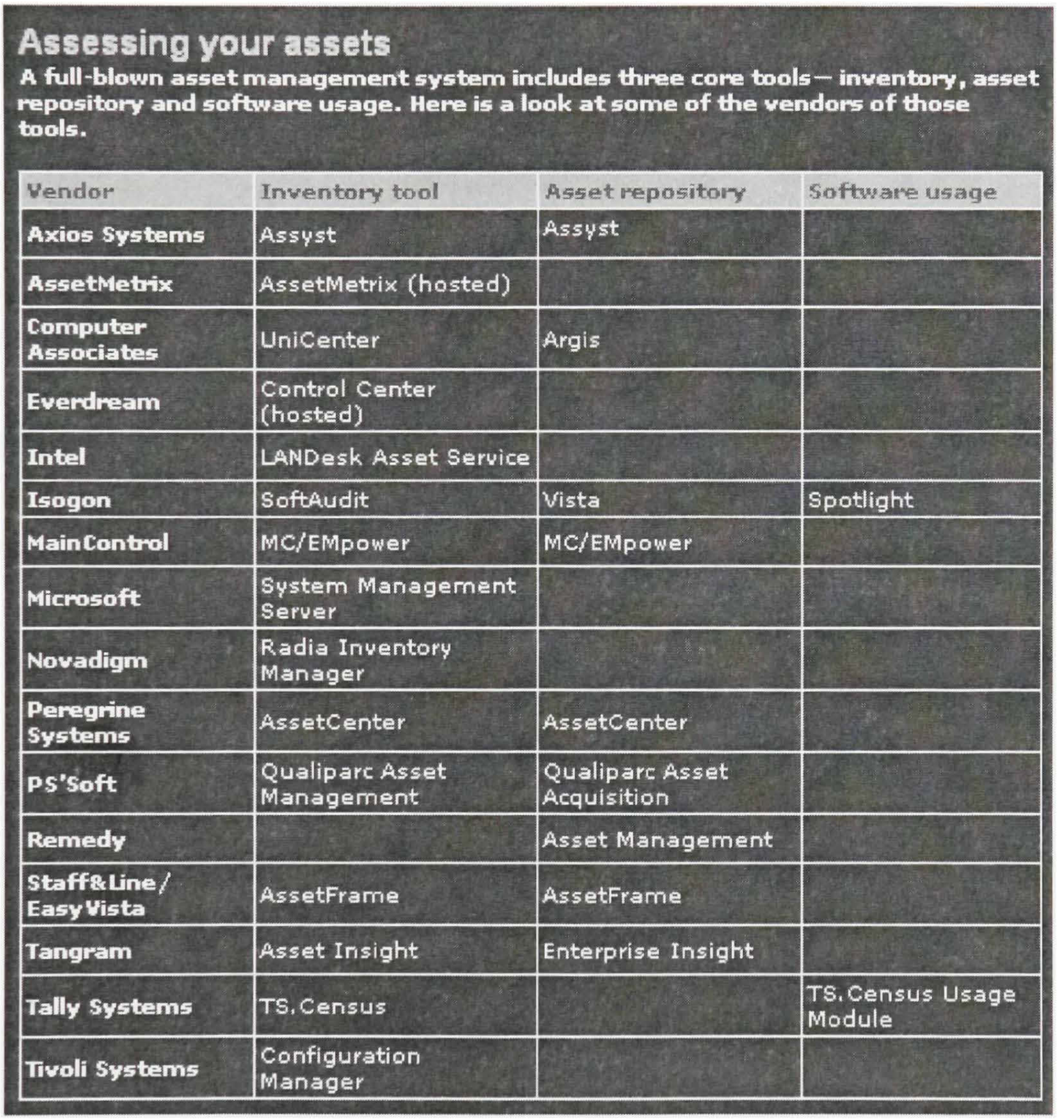
1. ABC Enterprise Systems
2. Absolute Software
3. Add-On Data
4. Adminpal
5. Alloy Software, Inc.
6. Altiris
7. Anthony Data
8. AppStream
9. ASAP eSMART
10. AssetMetrix
11. Astria Industries, Inc.
12. Atrium Technologies
13. Attest Systems, Inc.
14. AuditWizard
15. Automatos Inc.
16. Bentor Technologies
17. Blazent
18. Centennial Software
19. control.IT
20. Countermeasures Information Security, Inc.
21. Cynthia Farren Consulting
22. Elements Group
23. Endurics
24. Engagent Inc.
25. Everdream Corporation
26. Express Metrix
27. E-Z Audit
28. FileWave (USA) Inc.
29. GLOBEtrotter
30. Ibis,Inc.
31. IBM - License Management on Demand
32. INSYSTEK
33. Integrated Auditor™
34. Intel® LANDesk® Asset Service
35. Intellus Technology Management Solutions
36. International Association of IT Asset Managers
37. Intraware

38. Isotag Technology, Inc.
39. Komodo Digital
40. License Technologies Group, Inc.
41. LOGINventory
42. Maintenance Connection
43. ManageSoft
44. Marimba, Inc.
45. Mariner Tec Inc.
46. Micromation
47. mWired
48. MRO Software
49. netquartz
50. NetWorth Systems, Inc.
51. PcProfile
52. PC On Call
53. Peregrine Systems
54. Protexis Inc.
55. RPR Wyatt, Inc.
56. Sassafras Software Inc.
57. Scalable Software, Inc.
58. SchlumbergerSema
59. ShieldIP
60. Software Asset Management Services, Inc.
61. Sitekeeper
62. Smarte Solutions
63. Softricity, Inc.
64. Software Spectrum
65. Somix Technologies
66. Staff & Line
67. Tally Systems
68. Tangram Enterprise Solutions, Inc.
69. Techtracker ITX
70. Techserv USA
71. Touchpaper
72. TrackBird
73. xAssets
74. XML Alliances
75. Xpert Client Software

The industry is providing several automated solutions to the management of management of software assets.

APPENDIX III

SAM TOOL CLASSIFICATION



Assessing your assets

A full-blown asset management system includes three core tools— inventory, asset repository and software usage. Here is a look at some of the vendors of those tools.

Vendor	Inventory tool	Asset repository	Software usage
Axios Systems	Assyst	Assyst	
AssetMetrix	AssetMetrix (hosted)		
Computer Associates	UniCenter	Argis	
Everdream	Control Center (hosted)		
Intel	LANDesk Asset Service		
Isogon	SoftAudit	Vista	Spotlight
MainControl	MC/EMpower	MC/EMpower	
Microsoft	System Management Server		
Novadigm	Radia Inventory Manager		
Peregrine Systems	AssetCenter	AssetCenter	
PS'Soft	Qualiparc Asset Management	Qualiparc Asset Acquisition	
Remedy		Asset Management	
Staff&Line/ EasyVista	AssetFrame	AssetFrame	
Tangram	Asset Insight	Enterprise Insight	
Tally Systems	TS.Census		TS.Census Usage Module
Tivoli Systems	Configuration Manager		

Source:[18]

APPENDIX IV

STANDARDS AND/OR GUIDES OF JTC1/SC7 AS OF MARCH 2007 [65]

01 ISO 3535:1977	Forms design sheet and layout chart
02 ISO 5806:1984	Information processing -- Specification of single-hit decision tables
03 ISO 5807:1985	Information processing -- Documentation symbols and conventions for data, program and system flowcharts, program network charts and system resources charts
04 ISO/IEC 6592:2000	Information technology -- Guidelines for the documentation of computer-based application systems
05 ISO 6593:1985	Information processing -- Program flow for processing sequential files in terms of record groups
06 ISO/IEC 8631:1989	Information technology -- Program constructs and conventions for their representation
07 ISO 8790:1987	Information processing systems -- Computer system configuration diagram symbols and conventions
08 ISO 8807:1989	Information processing systems -- Open Systems Interconnection -- LOTOS -- A formal description technique based on the temporal ordering of observational behaviour
09 ISO/IEC 9126-1:2001	Software engineering -- Product quality -- Part 1: Quality model
10 ISO/IEC TR 9126-2:2003	Software engineering -- Product quality -- Part 2: External metrics
11 ISO/IEC TR 9126-3:2003	Software engineering -- Product quality -- Part 3: Internal metrics
12 ISO/IEC TR 9126-4:2004	Software engineering -- Product quality -- Part 4: Quality in use metrics
13 ISO 9127:1988	Information processing systems -- User documentation and cover information for consumer software packages
14 ISO/IEC TR 9294:2005	Information technology -- Guidelines for the management of software documentation
15 ISO/IEC 10746-1:1998	Information technology -- Open Distributed Processing -- Reference model: Overview
16 ISO/IEC 10746-2:1996	Information technology -- Open Distributed Processing -- Reference Model: Foundations

17 ISO/IEC 10746-3:1996	Information technology -- Open Distributed Processing -- Reference Model: Architecture
18 ISO/IEC 10746-4:1998	Information technology -- Open Distributed Processing -- Reference Model: Architectural semantics
19 ISO/IEC 10746-4:1998/ Amd 1:2001	Computational formalization
20 ISO/IEC 11411:1995	Information technology -- Representation for human communication of state transition of software
21 ISO/IEC TR 12182:1998	Information technology -- Categorization of software
22 ISO/IEC 12207:1995	Information technology -- Software life cycle processes
23 ISO/IEC 12207:1995/ Amd 1:2002	
24 ISO/IEC 12207:1995/ Amd 2:2004	
25 ISO/IEC 13235-1:1998	Information technology -- Open Distributed Processing -- Trading function: Specification
26 ISO/IEC 13235-3:1998	Information technology -- Open Distributed Processing -- Trading Function -- Part 3: Provision of Trading Function using OSI Directory service
27 ISO/IEC 13235-3:1998/ Cor 1:2006	
28 ISO/IEC 14102:1995	Information technology -- Guideline for the evaluation and selection of CASE tools
29 ISO/IEC 14143-1:1998	Information technology -- Software measurement -- Functional size measurement -- Part 1: Definition of concepts
30 ISO/IEC 14143-2:2002	Information technology -- Software measurement -- Functional size measurement -- Part 2: Conformity evaluation of software size measurement methods to ISO/IEC 14143-1:1998
31 ISO/IEC TR 14143-3:2003	Information technology -- Software measurement -- Functional size measurement -- Part 3: Verification of functional size measurement methods

32 ISO/IEC TR 14143-4:2002	Information technology -- Software measurement -- Functional size measurement -- Part 4: Reference model
33 ISO/IEC TR 14143-5:2004	Information technology -- Software measurement -- Functional size measurement -- Part 5: Determination of functional domains for use with functional size measurement
34 ISO/IEC 14143-6:2006	Information technology -- Software measurement -- Functional size measurement -- Part 6: Guide for use of ISO/IEC 14143 series and related International Standards
35 ISO/IEC TR 14471:1999	Information technology -- Software engineering -- Guidelines for the adoption of CASE tools
36 ISO/IEC 14568:1997	Information technology -- DXL: Diagram eXchange Language for tree-structured charts
37 ISO/IEC 14598-1:1999	Information technology -- Software product evaluation -- Part 1: General overview
38 ISO/IEC 14598-2:2000	Software engineering -- Product evaluation -- Part 2: Planning and management
39 ISO/IEC 14598-3:2000	Software engineering -- Product evaluation -- Part 3: Process for developers
40 ISO/IEC 14598-4:1999	Software engineering -- Product evaluation -- Part 4: Process for acquirers
41 ISO/IEC 14598-5:1998	Information technology -- Software product evaluation -- Part 5: Process for evaluators
42 ISO/IEC 14598-6:2001	Software engineering -- Product evaluation -- Part 6: Documentation of evaluation modules
43 ISO/IEC 14750:1999	Information technology -- Open Distributed Processing -- Interface Definition Language
44 ISO/IEC 14752:2000	Information technology -- Open Distributed Processing -- Protocol support for computational interactions
45 ISO/IEC 14753:1999	Information technology -- Open Distributed Processing -- Interface references and binding
46 ISO/IEC 14756:1999	Information technology -- Measurement and rating of performance of computer-based software systems

47 ISO/IEC TR 14759:1999	Software engineering -- Mock up and prototype -- A categorization of software mock up and prototype models and their use
48 ISO/IEC 14764:2006	Software Engineering -- Software Life Cycle Processes -- Maintenance
49 ISO/IEC 14769:2001	Information technology -- Open Distributed Processing -- Type Repository Function
50 ISO/IEC 14771:1999	Information technology -- Open Distributed Processing -- Naming framework
51 ISO/IEC 15026:1998	Information technology -- System and software integrity levels
52 ISO/IEC TR 15271:1998	Information technology -- Guide for ISO/IEC 12207 (Software Life Cycle Processes)
53 ISO/IEC 15288:2002	Systems engineering -- System life cycle processes
54 ISO/IEC 15289:2006	Systems and software engineering -- Content of systems and software life cycle process information products (Documentation)
55 ISO/IEC 15414:2006	Information technology -- Open distributed processing -- Reference model -- Enterprise language
56 ISO/IEC 15437:2001	Information technology -- Enhancements to LOTOS (E-LOTOS)
57 ISO/IEC 15474-1:2002	Information technology -- CDIF framework -- Part 1: Overview
58 ISO/IEC 15474-2:2002	Information technology -- CDIF framework -- Part 2: Modelling and extensibility
59 ISO/IEC 15475-1:2002	Information technology -- CDIF transfer format -- Part 1: General rules for syntaxes and encodings
60 ISO/IEC 15475-2:2002	Information technology -- CDIF transfer format -- Part 2: Syntax SYNTAX.1
61 ISO/IEC 15475-3:2002	Information technology -- CDIF transfer format -- Part 3: Encoding ENCODING.1
62 ISO/IEC 15476-1:2002	Information technology -- CDIF semantic metamodel -- Part 1: Foundation
63 ISO/IEC 15476-2:2002	Information technology -- CDIF semantic metamodel -- Part 2: Common
64 ISO/IEC 15476-3:2006	Information technology -- CDIF semantic metamodel -- Part 3: Data definitions
65 ISO/IEC 15476-4:2005	Information technology -- CDIF semantic metamodel -- Part 4: Data models

66 ISO/IEC 15476-6:2006	Information technology -- CDIF semantic metamodel -- Part 6: State/event models
67 ISO/IEC 15504-1:2004	Information technology -- Process assessment -- Part 1: Concepts and vocabulary
68 ISO/IEC 15504-2:2003	Information technology -- Process assessment -- Part 2: Performing an assessment
69 ISO/IEC 15504-2:2003/ Cor 1:2004	
70 ISO/IEC 15504-3:2004	Information technology -- Process assessment -- Part 3: Guidance on performing an assessment
71 ISO/IEC 15504-4:2004	Information technology -- Process assessment -- Part 4: Guidance on use for process improvement and process capability determination
72 ISO/IEC 15504-5:2006	Information technology -- Process Assessment -- Part 5: An exemplar Process Assessment Model
73 ISO/IEC TR 15846:1998	Information technology -- Software life cycle processes -- Configuration Management
74 ISO/IEC 15909-1:2004	Software and system engineering -- High-level Petri nets -- Part 1: Concepts, definitions and graphical notation
75 ISO/IEC 15910:1999	Information technology -- Software user documentation process
76 ISO/IEC 15939:2002	Software engineering -- Software measurement process
77 ISO/IEC 15940:2006	Information Technology -- Software Engineering Environment Services
78 ISO/IEC 16085:2004	Information technology -- Software life cycle processes -- Risk management
79 ISO/IEC TR 16326:1999	Software engineering -- Guide for the application of ISO/IEC 12207 to project management
80 ISO/IEC 18019:2004	Software and system engineering -- Guidelines for the design and preparation of user documentation for application software
81 ISO/IEC 19500-2:2003	Information technology -- Open Distributed Processing -- Part 2: General Inter-ORB Protocol (GIOP)/Internet Inter-ORB Protocol (IIOP)
82 ISO/IEC 19501:2005	Information technology -- Open Distributed Processing -- Unified Modeling Language (UML) Version 1.4.2

83 ISO/IEC TR 19759:2005	Software Engineering -- Guide to the Software Engineering Body of Knowledge (SWEBOK)
84 ISO/IEC TR 19760:2003	Systems engineering -- A guide for the application of ISO/IEC 15288 (System life cycle processes)
85 ISO/IEC 19761:2003	Software engineering -- COSMIC-FFP -- A functional size measurement method
86 ISO/IEC 19770-1:2006	Information technology -- Software asset management -- Part 1: Processes
87 ISO/IEC 20000-1:2005	Information technology -- Service management -- Part 1: Specification
88 ISO/IEC 20000-2:2005	Information technology -- Service management -- Part 2: Code of practice
89 ISO/IEC 20926:2003	Software engineering -- IFPUG 4.1 Unadjusted functional size measurement method -- Counting practices manual
90 ISO/IEC 20968:2002	Software engineering -- Mk II Function Point Analysis -- Counting Practices Manual
91 ISO/IEC 23026:2006	Software Engineering -- Recommended Practice for the Internet -- Web Site Engineering, Web Site Management, and Web Site Life Cycle
92 ISO/IEC 24570:2005	Software engineering -- NESMA functional size measurement method version 2.1 -- Definitions and counting guidelines for the application of Function Point Analysis
93 ISO/IEC 25000:2005	Software Engineering -- Software product Quality Requirements and Evaluation (SQuaRE) -- Guide to SQuaRE
94 ISO/IEC 25051:2006	Software engineering -- Software product Quality Requirements and Evaluation (SQuaRE) -- Requirements for quality of Commercial Off-The-Shelf (COTS) software product and instructions for testing
95 ISO/IEC 25062:2006	Software engineering -- Software product Quality Requirements and Evaluation (SQuaRE) -- Common Industry Format (CIF) for usability test reports
96 ISO/IEC 90003:2004	Software engineering -- Guidelines for the application of ISO 9001:2000 to computer software

APPENDIX V

IEEE STANDARDS ON SOFTWARE ENGINEERING AS OF MARCH [66]

	Standard Number
01	IEEE Std 1028-1988 IEEE standard for software reviews and audits
02	IEEE Std 1012-1986 IEEE standard for software verification and validation plans
03	IEEE Std 1063-1987 IEEE standard for software user documentation
04	IEEE Std 1058.1-1987 IEEE standard for software project management plans
05	ANSI/IEEE Std 1042-1987 IEEE guide to software configuration management
06	IEEE Std 1016-1987 IEEE Recommended Practice for Software Design Descriptions
07	ANSI/IEEE Std 1008-1987 IEEE standard for software unit testing
08	IEEE Std 1002-1987 IEEE standard taxonomy for software engineering standards
09	IEEE Std 1016.1-1993 IEEE guide to software design descriptions
10	IEEE Std 1061-1992 IEEE standard for a software quality metrics methodology
11	IEEE Std 1045-1992 IEEE standard for software productivity metrics
12	IEEE Std 1044-1993 IEEE standard classification for software anomalies.
13	IEEE Std 1059-1993 IEEE guide for software verification and validation plans
14	IEEE Std 1062-1993 IEEE recommended practice for software acquisition.
15	IEEE Std 1044.1-1995 IEEE guide to classification for software anomalies
16	IEEE 730-1989 IEEE Standard for Software Quality Assurance Plans
17	ANSI/IEEE Std 983-1986 An American National Standard - IEEE Guide For Software Quality Assurance Planning
18	IEEE Std 1028-1997 IEEE Standard for Software Reviews
19	IEEE Std 1012-1998 IEEE standard for software verification and validation
20	IEEE Std 1062, 1998 Edition IEEE recommended practice for software acquisition
21	IEEE Std 1058-1998 IEEE standard for software project management plans
22	IEEE Std 1012a-1998 Supplement to IEEE standard for software verification and validation: content map to IEEE/EIA 12207.1-1997
23	IEEE Std 1016-1998 IEEE recommended practice for software design descriptions
24	IEEE Std 1061-1998 IEEE standard for a software quality metrics methodology
25	IEEE Std 1012-2004 (Revision of IEEE Std 1012-1998) IEEE Std 1012 - 2004 IEEE Standard for Software Verificiation and Validation
26	IEEE Std 610.12-1990 IEEE standard glossary of software engineering terminology
27	IEEE Std 1219-1993 IEEE standard for software maintenance
28	IEEE Std 1298-1992; AS 3563.1-1991 Software quality management system. Part 1: requirements. Adopted from standards Australia IEEE Std 1074-1991 IEEE standard for developing software life cycle processes
29	IEEE Std 1228-1994 IEEE standard for software safety plans.
30	IEEE Std 1074-1995 IEEE standard for developing software life cycle processes
31	IEEE Std 1348-1995 IEEE recommended practice for the adoption of Computer-Aided Software Engineering (CASE) tools
32	IEEE Std 1420.1-1995 (IEEE standard for information technology - software reuse - dat... IEEE standard for information technology - software reuse - data model for reuse library interoperability: Basic Interoperability Data Model (BIDM)

33	IEEE Std 1387.2-1995 IEEE standard for information technology - Portable Operating System Interface (POSIX) system administration - part 2: software administration
34	IEEE Std 1074.1-1995 IEEE guide for developing software life cycle processes
35	IEEE Std 1430-1996 IEEE guide for information technology - software reuse - concept of operations for interoperating reuse libraries IEEE Std 1226.3-1998 IEEE standard for software interface for resource management for A Broad-Based Environment for Test (ABBET)
36	IEEE Std 1219-1998 IEEE standard for software maintenance
37	IEEE Std 730-1998 IEEE standard for software quality assurance plans
38	IEEE Std 1074-1997 IEEE standard for developing software life cycle processes
39	IEEE Std 1420.1b-1999 IEEE trial-use supplement to IEEE standard for information technology - software reuse - data model for reuse library interoperability: intellectual property rights framework
40	IEEE Std 14143.1-2000 Implementation note for IEEE adoption of ISO/IEC 14143-1:1998. Information technology - software measurement - functional size measurement. Part 1: definition of concepts
41	IEEE Std 1471-2000 IEEE Recommended practice for architectural description of software-intensive systems
42	IEEE Std 1540-2001 IEEE Standard for Software Life Cycle Processes - Risk Management
43	IEEE Std 1063-2001 IEEE standard for software user documentation
44	IEEE Std 730-2002 (Revision of IEEE Std 730-1998) IEEE Standard for Software Quality Assurance Plans
45	IEEE Std 1175.3-2004 IEEE Standard for CASE Tool Interconnections-Reference Model for Specifying Software Behavior
46	IEEE Std 1558-2004 IEEE Standard for Software Documentation for Rail Equipment and Systems
47	IEEE Std 1517-1999(R2004) IEEE Standard for Information Technology - Software Life Cycle Processes - Reuse Processes
48	IEEE Std 1074-2006 (Revision of IEEE Std 1074-1997) IEEE Standard for Developing a Software Project Life Cycle Process
49	IEEE Std 830-1984 IEEE guide to software requirements specifications
50	IEEE Std 828-1990 IEEE standard for software configuration management plans
51	IEEE Std 982.2-1988 IEEE guide for the use of IEEE standard dictionary of measures to produce reliable software
52	IEEE Std 982.1-1988 IEEE standard dictionary of measures to produce reliable software
53	IEEE Std 730.1-1989 IEEE standard for software quality assurance plans
54	IEEE Std 830-1993 IEEE recommended practice for software requirements specifications
55	IEEE Std 730.1-1995 IEEE guide for software quality assurance planning
56	IEEE Std 829-1983 IEEE Standard For Software Test Documentation
57	IEEE/EIA 12207.1-1997 Industry implementation of International Standard ISO/IEC 12207: 1995. (ISO/IEC 12207) standard for information technology - software life cycle processes - life cycle data
58	IEEE/EIA 12207.2-1997 Industry implementation of International Standard ISO/IEC 12207: 1995. (ISO/IEC 12207 standard for information technology - software life cycle processes - implementation considerations
59	IEEE/EIA 12207.0-1996 IEEE/EIA 12207.0-1996 IEEE/EIA Standard Industry Implementation of International Standard ISO/IEC 12207: 1995 (ISO/IEC 12207) Standard for Information Technology Software Life Cycle Processes
60	IEEE Std 828-1998 IEEE Standard For Software Configuration Management Plans
61	IEEE Std 830-1998 IEEE recommended practice for software requirements specifications
62	IEEE Std 829-1998 IEEE standard for software test documentation
	J-STD-016-1995 Trial-use standard standard for information technology software life cycle processes software development acquirer-supplier agreement
63	IEEE Std 828-2005 (Revision of IEEE Std 828-1998) IEEE Standard for Software Configuration Management Plans

64	IEEE Std 982.1-2005 (Revision of IEEE Std 982.1-1988) IEEE Std 982.1 - 2005 IEEE Standard Dictionary of Measures of the Software Aspects of Dependability
65	ISO/IEC 23026; IEEE Std 2001-2002, First edition Software Engineering - Recommended Practice for the Internet - Web Site Engineering, Web Site Management, and Web Site Life Cycle Ref : http://ieeexplore.ieee.org/xpl/standards.jsp?letter=software&type=2&count=68&findtitle=software&page=2&ResultStart=50 on October 9th, 2006.

APPENDIX VI

ITIL DEFINITIONS

Table 18

ITIL goals by process [14, 15]

Process	Goal
Incident Management	The primary goal of the Incident Management process is to **restore normal service operation as quickly as possible** and minimize the adverse impact on business operations
Problem Management	The goal of Problem Management is to **minimize the adverse impact of Incidents and Problems on the business** that are caused by errors within the IT Infrastructure, and to prevent recurrence of Incidents related to these errors
Configuration Management	**Account for all the IT assets and configurations within the organization** and its services provide accurate information on configurations and their documentation to support all the other Service Management processes provide a sound basis for Incident Management, Problem Management, Change Management and Release Management verify the configuration records against the infrastructure and correct any exceptions.
Change Management	The goal of the Change Management process is to ensure that standardized methods and procedures are used for **efficient and prompt handling of all Changes, in order to minimize the impact of Change-related Incidents upon service quality**, and consequently to improve the day-to-day operations of the organization.
Release Management	**To plan and oversee the successful rollout of software and related hardware**
Service Level Management	**The goal for SLM is to maintain and improve IT Service quality, through a constant cycle of agreeing, monitoring and reporting** upon IT Service achievements and instigation of actions to eradicate poor service - in line with business or Cost justification
Financial Management	**To be able to account fully for the spend on IT Services** and to attribute these costs to the services delivered to the organisation's Customers. To assist management decisions on IT investment by providing detailed business cases for Changes to IT Services.
IT Service	**To support the overall Business Continuity Management process**

Process	Goal
Continuity Management	**by ensuring that the required IT technical and services facilities can be recovered within required, and agreed, business timescales**.
Capacity Management	**Capacity Management needs to understand the business requirements** (the required Service Delivery), the organization's operation (the current Service Delivery) and the IT Infrastructure (the means of Service Delivery), **and ensure that all the current and future Capacity and performance aspects of the business requirements are provided cost-effectively**.
Availability Management	To optimize the capability of the IT Infrastructure, services and supporting organization to deliver a Cost effective and sustained level of Availability that enables the business to satisfy its business objectives

APPENDIX VII

ITIL AND ASSET MANAGEMENT

The following table lists all occurrences of the term “Asset” in ITIL Service Support and Service Development books.

Table 19

Usage of "Asset" wihtin ITIL Service support

Quote	Comment
“Given the definition above, it should be clear that Configuration Management is not synonymous with Asset Management, although the two disciplines are related. Asset Management is a recognised accountancy process that includes depreciation accounting. Asset Management systems maintain details on assets above a certain value, their business unit and their location. Configuration Management also maintains relationships between assets, which Asset Management usually does not. Some organisations start with Asset Management and then move on to Configuration Management. ”	
● “The basic activities of Configuration Management are as follows: ○ […] ○ Identification. Selecting and identifying the configuration structures for all the infrastructure's CIs, including their 'owner', their interrelationships and configuration documentation. It includes allocating identifiers and version numbers for CIs, labelling each item, and entering it on the Configuration Management Database (CMDB). ○ […] ○ Status accounting. […] CIs and their records to be traceable, e.g. tracking the status of a CI as it changes from one state to another for instance 'under development', 'being tested', 'live', or 'withdrawn'. ○ Verification and audit. A series of reviews and audits that verify the physical existence of CIs and check that they are correctly recorded in the Configuration Management system. ○ […].”	
○ “Do you have the Service Management and business expertise to design: [..] integration with other support tools such as Asset Management and Configuration Management, Change control and automated operations?”	And when talking about the Service Desk:

Quote	Comment
o "7.5 Planning and implementation o Many enterprises implement Asset Management before implementing Configuration Management. The processes in this section apply to both Asset Management and Configuration Management."	
o And of the cost of IT finance management: "2.9 Financial Management for IT Services o Financial Management is responsible for accounting for the costs of providing IT service and for any aspects of recovering these costs from the Customers (charging). It requires good interfaces with Capacity Management, Configuration Management (asset data) and Service Level Management to identify the true costs of service."	
Although license management is mentioned in ITIL in section 7.3.10 License management: o "Company directors, senior managers, and others, are liable to face imprisonment and fines if illegal software is found to be in use within their enterprise. Configuration Management enables an enterprise to monitor and control software licences, from purchase to disposal. Software licence structures, and corporate and multi-licensing schemes, need to be understood and communicated to service-provider staff and Customers. o Responsibility for controlling and auditing software licences should be unambiguous and should involve purchasing and Asset or Configuration Management. This may be difficult when Users find it so easy to purchase and download software from the Internet, but this can be resolved by links to disciplinary procedures detailed within the organisation's Security Policy"	
o As for Roles Asset Manager is a separate role but not explained o "Roles: The roles within the function should include Configuration Manager, Asset Manager, Change Manager, Change administrator, Release Manager and relevant Change Advisory Board(s). " o Release management mentions that asset may require a tag; not nothing else is mentioned: "9.6.4 Rollout planning o Rollout planning extends the Release plan produced so far to add details of the exact installation process developed and the agreed implementation plan. Rollout planning involves:	

Quote	Comment
▪ producing an exact, detailed timetable of events, as well as who will do what (i.e. a resource plan) ▪ listing the CIs to install and decommission, with details on the method of disposal for any redundant equipment and software documenting an action plan by site, noting any implications of different time zones on the overall plans (e.g. an international organisation may well not have a single common Release window when none of its systems is being used throughout the world) ▪ producing Release notes and communications to end Users ▪ Planning communication ▪ developing purchasing plans ▪ acquiring hardware and software where, because this often involves the acquisition and deployment of numerous high-value assets, the rollout plan should include the procedures to be followed for their secure storage prior to rollout and the mechanisms to trace their deployment during the implementation (which could involve the use of asset tags or other electronically readable labels) ▪ scheduling meetings for managing staff and groups involved in the Release.”	

Table 20

Usage of "Asset" within the ITIL Service Development book

Quote	Comment
In Service Delivery, an asset is defined as ▪ “Component of a business process. Assets can include people, accommodation, computer systems, networks, paper records, fax machines, etc.” In section “5.1.5 Relationship with other IT Service Management processes”, ITIL mentions “Configuration Management -Financial Management requires Asset and cost information that may be managed by large organisation-wide systems. Configuration Management is responsible for managing the data relating to assets (Configuration items) and their attributes (e.g. cost)”	

APPENDIX VIII

THE AMALGAMATION OF EXISTING STANDARDS TO DEFINE SAM PROCESSES

The amalgamation of organizational processes

The following table is an amalgamation of organizational processes taken from IIMM, ITIL, BS15000, SWEBOK, ISO19770-1, ISO 15939 and IEEE Std 1062 that fulfills the needs mentioned in **Table 10**.

Table 21

Organizational processes

Goal	Source	Process grouping	Addressed by Process
Make corporate management aware on the benefits of AM and the associated policy and financial implication	IIMM	Corporate AM Direction	Obtain organizational commitment
Impart ownership of the process by involving the key players in the AM programme.	IIMM	Corporate AM Direction	Obtain organizational commitment
Communicate AM objectives to the staff	IIMM	Corporate AM Direction	Obtain organizational commitment
Ensure the staff are provided with the appropriate systems, training and resources.	IIMM	Corporate AM Direction	Obtain organizational commitment
Maintain the focus on AM at all levels be continued involvement of key staff and review process and inform of targets, progress and achievements.	IIMM	Corporate AM Direction	Obtain organizational commitment
To produce agreed, timely, reliable, accurate reports for informed decision making and effective communication.	BS 15000	Service delivery process	Service reporting
Accept the requirements for measurement and assign resources	ISO 15939		Establish and sustain measurement commitment

Table 21
Organizational processes (suite)

Goal	Source	Process grouping	Addressed by Process
Characterize organizational unit	ISO 15939		Plan the measurement process
Identify information needs	ISO 15939		Plan the measurement process
Select measures	ISO 15939		Plan the measurement process
Define data collection, analysis, and reporting procedures	ISO 15939		Plan the measurement process
Define criteria for evaluating the information products and the measurement process	ISO 15939		Plan the measurement process
Review, approve, and provide resources for measurement tasks	ISO 15939		Plan the measurement process
Acquire and deploy supporting technologies	ISO 15939		Plan the measurement process
Ensure that responsibility for management of software assets is recognized	ISO/IEC 19770-1	4.2 Control Environement for SAM	Corporate Governance
Roles and responsibilities for software and related assets are clearly defined, maintained and understood	ISO/IEC 19770-1	4.2 Control Environement for SAM	Roles and Responsability
Organization maintains clear policies, processes and procedures to ensure effective planning, operation and control of SAM	ISO/IEC 19770-1	4.2 Control Environement for SAM	Policies processes and procedures
Appropriate competence and expertise in SAM is available and is being applied	ISO/IEC 19770-1	4.2 Control Environement for SAM	Competence in SAM

Table 21
Organizational processes (suite)

Goal	Source	Process grouping	Addressed by Process
Appropriate preparation and planning for the effective and efficient accomplishment of SAM objectives	ISO/IEC 19770-1	4.3 Planning and Implementation Processes for SAM	Planning for SAM
Accomplish overall SAM objectives and the SAM plan	ISO/IEC 19770-1	4.3 Planning and Implementation Processes for SAM	Implementation for SAM
Ensure that the management objectives for SAM are being achieved	ISO/IEC 19770-1	4.3 Planning and Implementation Processes for SAM	Monitoring and review for SAM
Ensure that opportunities for improvement are identified and acted upon	ISO/IEC 19770-1	4.3 Planning and Implementation Processes for SAM	Continual improvement for SAM
Ensure that the necessary classes of assets are selected and grouped; and defined by appropriate characteristics that enable effective and efficient control	ISO/IEC 19770-1	4.3 Planning and Implementation Processes for SAM	Implementation for SAM
Information products and evaluation results in the "Experience Base" should be consulted during the performance of this activity.			

SAM specific requirements

The following table associates the SAM specific processes with processes from other sources that have similar goals.

Table 22

SAM specific needs

#	Process label	Process goal	Mapping	source
10	Define terms and conditions		Compliance	
11	Define SAM scope		SAM	
12	SAM standardization		SAM	
13	SAM planning and monitoring criteria		SAM	

Table 23

SAM specific processes

Goal	Source	Process grouping	Process
ensure that records reflect accurately and completely what they are supposed to record, and conversely that what they record has not changed without approval	ISO/IEC 19770-1	4.5 Verification and Compliance Processes for SAM	Software Asset Record Verification
ensure that all intellectual property used by the organization but owned by others, pertaining to software and related assets, is properly licensed and used in accordance with its terms and conditions	ISO/IEC 19770	4.5 Verification and Compliance Processes for SAM	Software Asset Licensing compliance
ensure that security requirements related to the use of software and related assets are complied with	ISO/IEC 19770	4.5 Verification and Compliance Processes for SAM	Software Asset Security compliance
ensure that there is continuing compliance with the requirements of this part of ISO/IEC 19770 including compliance with required policies and procedures	ISO/IEC 19770	4.5 Verification and Compliance Processes for SAM	Conformance verification for SAM
manage relationships with other organizations, both external and internal, to ensure the provision of seamless, quality SAM services, and to manage all contracts for software and related assets and services	ISO/IEC 19770	4.6 Operations Management Processes and Interfaces for SAM	Relationship and contract management for SAM

Goal	Source	Process grouping	Process
budgeting and accounting for software and related assets; and ensuring that relevant financial information is readily available for financial reporting, tax planning, and calculations	ISO/IEC 19770	4.6 Operations Management Processes and Interfaces For SAM	Financial Management for SAM
to define, record and manage levels of service related to SAM.	ISO/IEC 19770	4.6 Operations Management Processes and Interfaces For SAM	Service Level Management for SAM
to manage information security effectively within all SAM activities and support the approval requirements related to SAM	ISO/IEC 19770	4.6 Operations Management Processes and Interfaces For SAM	Security Management for SAM

Interface of processes requirements

The following table associates the interfaces to existing processes from other sources that have similar goals.

Table 24

Needs to interface with existing processes

#	Process label	Process goal	Mapping	source
02	Acquisition		Acquisition	
03	State indicator		Configuration management	
04	Baseline		Configuration management	
05	Quality Control		Quality Assurance	

Table 25

Interface with existing processes

Goal	Source	Process grouping	Process
The software acquisition life cycle represents the period of time that begins with the decision to acquire a software product and ends when the product is no longer available for use. It typically includes a planning phase, contracting phase, product implementation phase, product acceptance phase, and follow-on phase. This life cycle provides an overall framework within which most software acquisitions occur	IEEE Std 1062		Software acquisition process
ensure that they are acquired in a controlled manner and properly recorded	ISO/IEC 19770		Acquisition Process
ensure that all changes which impact on SAM are assessed, approved, implemented and reviewed in a controlled manner and meet all record-keeping requirements	ISO/IEC 19770-1	4.7 Life Cycle Interfaces for SAM	Change Management Process
To ensure all changes are assessed, approved, implemented and reviewed in a controlled manner.	BS15000	Control processes	Change management
To define and control the components of the service and infrastructure and maintain accurate configuration information	BS15000	Control processes	Configuration management
Account for all the IT assets and configurations within the organization and its services provide accurate information on configurations and their documentation to support all the other processes	ITIL	Service support	Configuration Management
ensure that they are developed in a way which considers SAM requirements	ISO/IEC 19770	4.7 Life Cycle Interfaces for SAM	Software development process
ensure that releases are planned and executed in a way which supports SAM requirements	ISO/IEC 19770	4.7 Life Cycle Interfaces for SAM	Software Release Management process
To deliver, distribute and track one or more changes in a release into the live environment.	BS15000	Release process	Release management process
ensure that software deployment and redeployment is executed in a way which supports SAM	ISO/IEC 19770	4.7 Life Cycle Interfaces	Software Deployment process

Goal	Source	Process grouping	Process
requirements		for SAM	
monitor and respond to incidents in ongoing operations relevant to software and related assets	ISO/IEC 19770	4.7 Life Cycle Interfaces for SAM	Incident Management Process
To restore agreed service to the business as soon as possible or to respond to service requests.	BS15000	Resolution processes	Incident management
Keep software assets current and in operational fitness, including through proactive identification and analysis of the cause of incidents and addressing the underlying problems	ISO/IEC 19770	4.7 Life Cycle Interfaces for SAM	Problem Management process
To minimize disruption to the business by proactive identification and analysis of the cause of service incidents and by managing problems to closure	BS15000	Resolution processes	Problem management
remove software and related assets from use, including recycling of associated assets where appropriate, in accordance with company policy and meeting all record-keeping requirements	ISO/IEC 19770	4.7 Life Cycle Interfaces for SAM	Retirement Process

APPENDIX IX

GLOSSARY OF SAM RELATED TERMS

Table 26

Glossary of SAM related terms

Term	Definition	Used in
Asset	Component of a business process. Assets can include people, accommodation, computer systems, networks, paper records, fax machines, etc.	ITIL
Baseline	A snapshot or a position which is recorded. Although the position my be updated later, the baseline remains unchanged and available as a reference of the original state and as a comparison against the current position.	PRINCE2
Baselining	Process by which the quality and cost-effectiveness of a service is assessed, usually in advance of a change to the service. Baselining usually includes comparison of the service before and after the change or analysis of trend information. The term benchmarking is usually used if a comparison is made against other enterprises.	
Business process	A group of business activities undertaken by an organization in pursuit of a common goal. Typical business processes include receiving orders, marketing services, selling products, delivering services, distributing products, invoicing services, accounting for money received. A business process rarely operates in isolation, i.e. other processes will depend in it and it will depend on other processes.	
Capital costs		
SAM		IITL SAM book ISO/IEC 19770-1 SAM processes
Establish and Maintain	When using a CMMI model, you will encounter goals and practices that include the phrase "establish and maintain." This phrase connotes a meaning beyond the component terms; it includes documentation and usage. For example,	CMMI

Term	Definition	Used in
	"Establish and maintain an organizational policy for planning and performing the organizational process focus process" means that not only must a policy be formulated, but it also must be documented and it must be used throughout the organization	
Customer	A "customer" is the party (individual, project, or organization) responsible for accepting the product or for authorizing payment. The customer is external to the project, but not necessarily external to the organization. The customer may be a higher level project. Customers are a subset of stakeholders.	CMMI
Stakeholder	A "stakeholder" is a group or individual that is affected by or in some way accountable for the outcome of an undertaking. Stakeholders may include project members, suppliers, customers, end users, and others.	CMMI
Process	A "process," as used in the CMMI Product Suite, consists of activities that can be recognized as implementations of practices in a CMMI model. These activities can be mapped to one or more practices in CMMI process areas to allow a model to be useful for process improvement and process appraisal. (In Chapter 2, see the definition of "process area" and a description of how this term is used in the CMMI Product Suite.)	CMMI
Managed Process	A "managed process" is a performed process that is planned and executed in accordance with policy; employs skilled people having adequate resources to produce controlled outputs; involves relevant stakeholders; is monitored, controlled, and reviewed; and is evaluated for adherence to its process description.	CMMI
Defined Process	A "defined process" is a managed process that is tailored from the organization's set of standard processes according to the organization's tailoring guidelines; has a maintained process description; and contributes work products, measures, and other process-improvement information to the organizational process assets. (In Chapters 2 and 4, see the descriptions of how "defined process" is used in the CMMI Product Suite.) [FM114.HDA103.HDB108.T101] A project's defined process provides a basis for planning, performing, and improving the project's tasks and activities. A project may have more than one defined	CMMI

Term	Definition	Used in
	process (for example, one for developing the product and another for testing the product).	
SAM baseline		
SA inventory		
SA Entitlement		
Software retirement		
Definitive Software Library (DSL)	A secure storage area where the physical copies of the purchased software are stored : "It contains the master copies of all controlled software in an organisation"	

From [67] "6.2 Configuration management process

The Configuration Management Process is a process of applying administrative and technical procedures throughout the software life cycle to: identify, define, and baseline software items in a system; control modifications and releases of the items; record and report the status of the items and modification requests; ensure the completeness, consistency, and correctness of the items; and control storage, handling, and delivery of the items.

NOTE When this process is employed on other software products or entities, the term "software item" is interpreted accordingly".

APPENDIX X

THE EVOLUTION OF ISO/IEC 19770-1 THROUGH ITS DOCUMENTS

Historical background

It was observed that in the market place two approaches where often used to perform software inventory:

- The first approach was to compare the number of licenses with the number of software deployed. The exercise consisted of matching the licensing documentation with the number of software deployed. One important assumption is that the licensing documentation could easily be found and that the deployed software include all the software present in the organization; it was the experience of the working group that this assumption was strong and could lead to missing several licenses, software or documentation.
- The second approach consisted of scanning the hardware for all installed software. This resulted in the accumulation of an important amount of data on systems, components, freeware and various irrelevant files. However, the exercise of determining what was relevant and was not as well as identifying what programs where installed was very time consuming.

It was believed that developing a universal SAM tag would help simplify the exercise of identifying software after a scan was performed. However, WG21 did not have enough resources to work both on the TAG and the processes at the same time: the decision was made early to start with the processes since this would bring a common definition of SAM and specify standardized terms and definitions for SAM.

The Busan-2002 (strawman) version (N2622)

WG21 observations

In 2001, WG21 made some observations that can be grouped under four categories:

- The organization: The working group observed that organizations did not cover license management in a formal way: they lacked a central point of control, they did not understand copyright laws, they did not have clear software code of practice policies and the budget for SAM was insufficient.
- The technical aspects: The technique used for software identification caused problems and it was a difficult and time consuming task to understand the result of reports resulting from this inventory process.
- The legal aspect: Laws vary from country to country which can complicate even further mergers and de-mergers of organizations. Furthermore, software manufacturers, such as Microsoft Sweden, were asking from 3 sources for proof of purchase: licensing documentation, purchase documentation from the accounting system and a global license database covering the entire organization; but not all manufacturers asked for the same things.
- Licensing types: Organization may adopt various schemes to manage their licenses, from site licenses with stringent controls to less stringent controls or even incremental site licenses. They can also be bulk purchase agreements, "pay as you go" and pre-installed software. All these types of agreements have to be considered and managed accordingly.

Requirements for SAM processes

To manage software assets and to take into consideration the observations of the previous section, processes will have to be developed. At this stage of the writing of the standard, these processes were not yet written; the following is a list of requirements that were at that time though of as important to address in order to manage software assets:

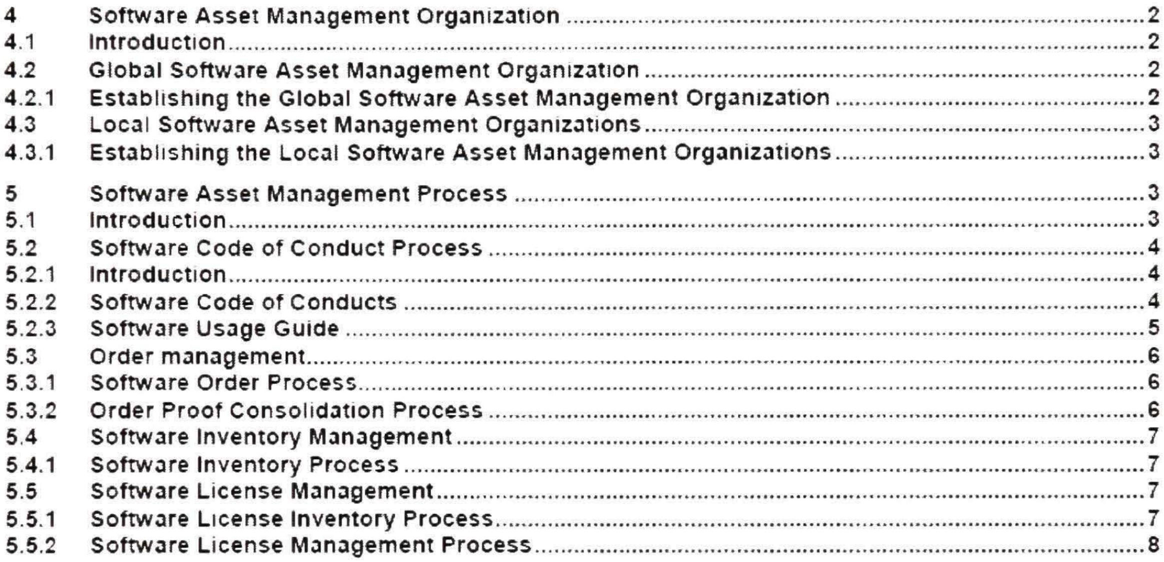
1. License Management
2. Asset Management Repository
3. Standard Application Catalog
4. License complexity
5. Inventory Management
6. Order Management
7. How to deal with software assets when companies merge?
8. How to deal with software assets when companies de-merge?
9. Global organizations
10. Different Software Asset Management roles.
11. Guideline for *Software Code of Conduct*
12. Interaction with help desk
13. Interaction with Change Management

In the Busan version of the standard, processes and documents are intertwined: for example, "Software Code of Conduct" is a document not a process. Even at this early stage, interfaces with ITIL processes can be seen with the interaction with the help desk and change management; both of which are found in ITIL. Furthermore, the third item, the standard service catalogue, is very much aligned with ITIL which recommends building a standard application catalogue to distinguish between standard and non-standard requests. Point numbers 7 and 8 were open questions about mergers and de-mergers; they do not constitute processes per say.

The WG21 panel of experts also discussed about the legal aspect of the terms and conditions of commercial software (APPENDIX XIII): to evaluate the legal impact of the terms and conditions, the legal department should be involved to identify potential risks for the organization.

This first version of the standard does not contain much text and process descriptions, but its structure and table of content provide a list of topics that will be discussed and voted upon in the following versions of the standard.

The Montreal-2003 version (N2885)



4 Software Asset Management Organization 2
4.1 Introduction 2
4.2 Global Software Asset Management Organization 2
4.2.1 Establishing the Global Software Asset Management Organization 2
4.3 Local Software Asset Management Organizations 3
4.3.1 Establishing the Local Software Asset Management Organizations 3

5 Software Asset Management Process 3
5.1 Introduction 3
5.2 Software Code of Conduct Process 4
5.2.1 Introduction 4
5.2.2 Software Code of Conducts 4
5.2.3 Software Usage Guide 5
5.3 Order management 6
5.3.1 Software Order Process 6
5.3.2 Order Proof Consolidation Process 6
5.4 Software Inventory Management 7
5.4.1 Software Inventory Process 7
5.5 Software License Management 7
5.5.1 Software License Inventory Process 7
5.5.2 Software License Management Process 8

Figure 19 Table of content of Montreal version -2003

Figure 19 presents a portion of the table of content from the Montreal version (N2885) of the standard. In section 4 of this version of the standard, the notion of local and international organization is introduced. This is to address the requirements and concerns brought up in the previous version of the document about licensing terms changing from one country to another; although a centralized organization must own the process, local laws and cultural differences must be taken into account separately by the local organization.

Section 5 of the standard (**Figure 19**) lists all the processes proposed for the management of Software Assets. The code of conduct is still present but has now two parts: the code of conduct and the usage guide: the code of conduct states what is expected from employees while the usage guide considers the fact that the guide will be used differently according to local needs (i.e. specifying local specificity).

Official comments

Official comments are also provided by other countries and these comments have to be addressed at the subsequent plenary meeting. Those comments are presented in APPENDIX XIII. For example, a comment from the Canadian representative consisted in a suggestion to include configuration management: licenses like their software counterparts, might exist under different versions and have to be managed as well. Furthermore, when performing an inventory, the current state of the inventory has to be captured and changes to it have to be monitored and managed.

Another comment made for the interim meeting of that same year, in APPENDIX XIII, came from the German delegation about adding roles and responsibilities to global and local organization so as to highlight the difference of responsibility between these two roles.

The Brisbane-2004 version (N3084)

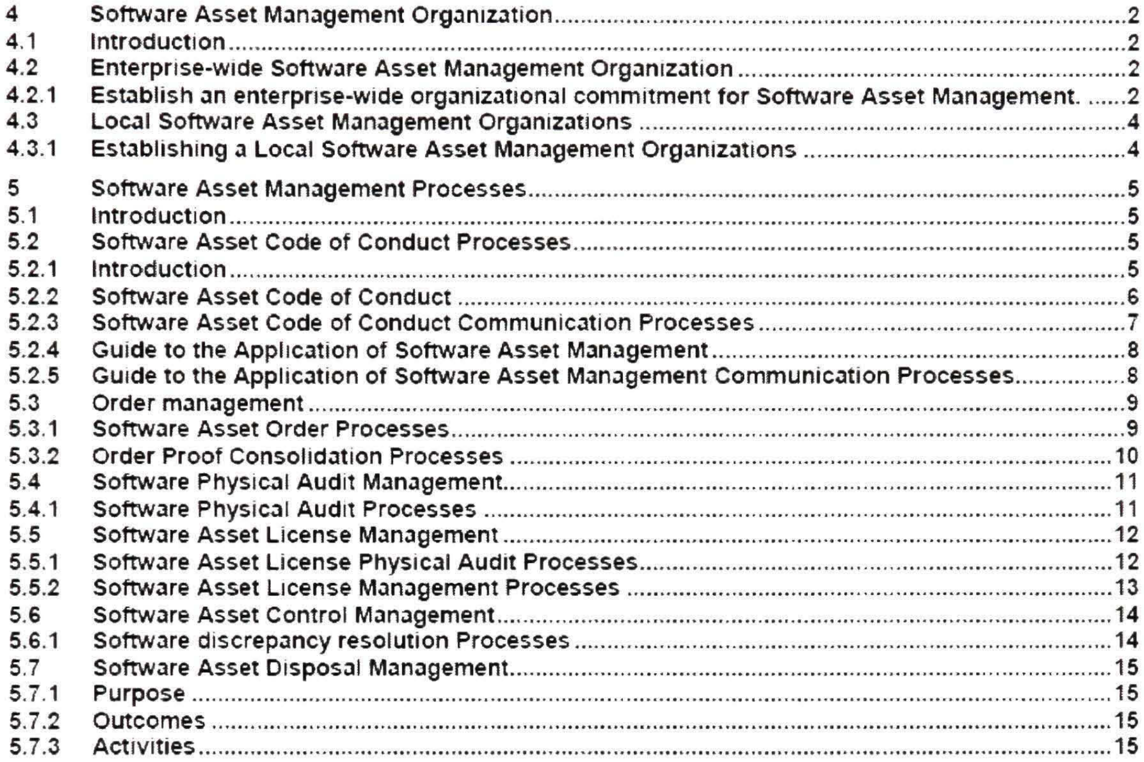


Figure 20 Table of content of the Brisbane version -2004

Figure 20 presents a portion of the table of content from the Brisbane version (N3084) of the standard. The code of conduct is still present but has been expended further: it still consists of a code of conduct (global organization) and a guide (local specificity) but it now has an "establishment section" to setup the process and a "communication section" to communicate it to the rest of the organization. This last section highlights the need to communicate the process within the organization. The guide to the application of SAM has also a very wide coverage: it not only explains how to use the asset order process, but also provides information on audits, the scanning process, and it describes different kinds of licenses and security policies.

Requirements for SAM processes

The software installed on hardware is now called the physical software which has to be managed together with the licenses and must be reconciled when discrepancies are observed. **Figure 20** illustrates the entire set of processes: it starts with corporate requirements which translate into having commitment from upper management and through policies and guidelines to better communicate the corporate requirements. Once corporate requirements are established, software assets budget and needs have to be planned, including planning for provisions for all the software assets that will be needed while budgeting for licenses that will be needed latter but no known at the moment of the planning process. **Figure 20**, also shows that software asset and licenses are two separate entities and are managed in parallel but must also match: the reconciliation process is done through the "Inventory Control Management" process.

Thursday, May 13th, 2004

Software Asset Management Processes Diagram

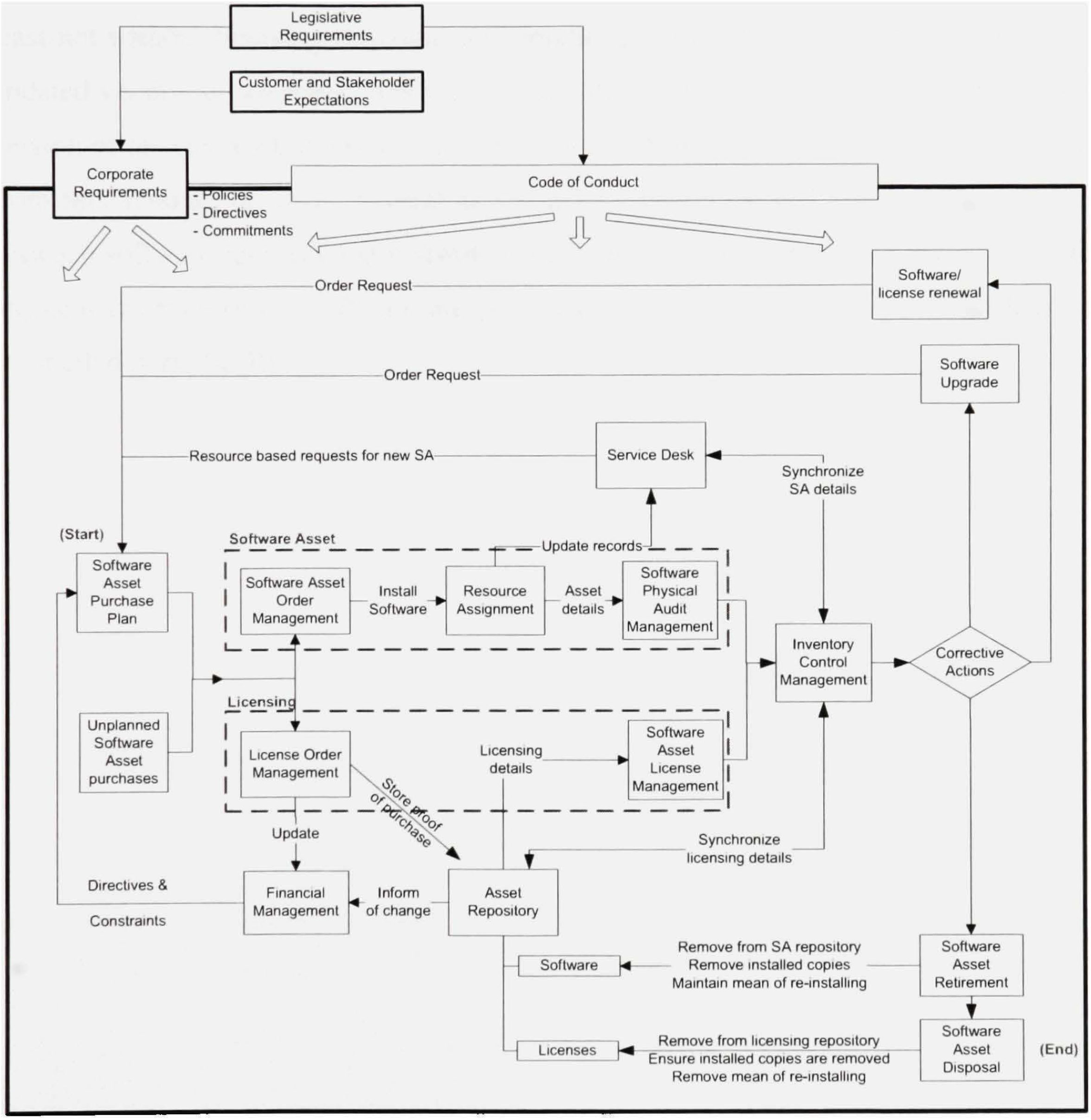


Figure 21 SAM lifecycle - ISO/IEC 19770-1 - Brisbane 2004 version

Figure 21 also introduces a new concept: the difference between software asset retirement and software asset disposal: a software might be removed from the IT infrastructure (not available to users) but may still be available to the organization; if licenses are not available anymore, the software is not available to the organization, at least not without breaking the terms and conditions of the license. Table 12 presents an updated version of the topics that are new and the topics that are not part of this current version of the standard. Interfaces to service desk, change management are still excluded from this version; however, several new concepts are introduced such as the difference between software removal and software retirement. The management of licenses and the physical instance of the software are performed in parallel but must be controlled and reconciled periodically.

The Helsinki-2005 version (N3276)

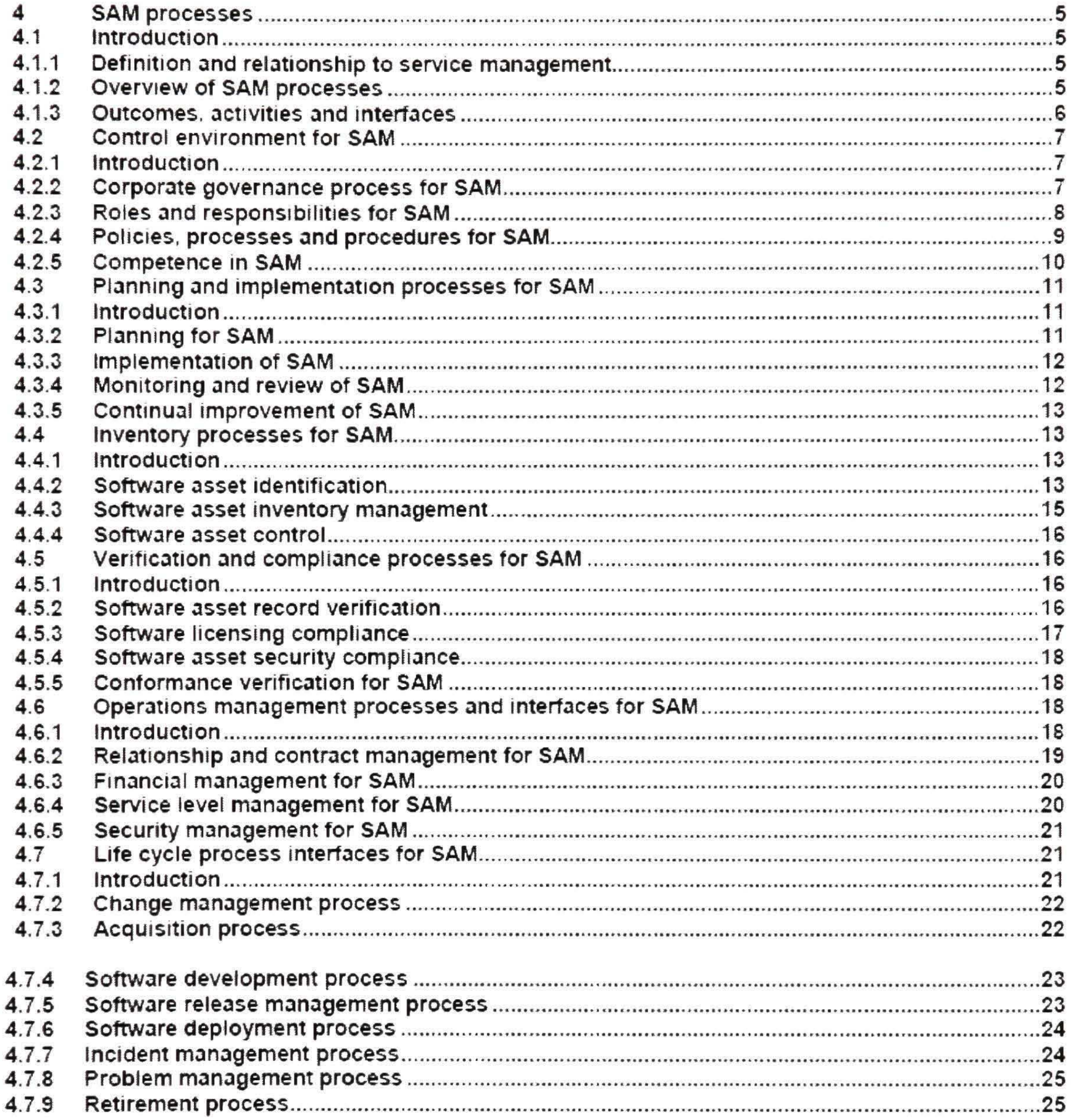


Figure 22 Table of Content of the Helsinki version - 2005

Indeed several processes from ISO/IEC 20000 (still BS1500 at the time), were mapped to ISO/IEC 19770-1, sometimes in the Primary processes section, sometimes in the Support processes. The following is a list of ISO/IEC 20000 processes that were adapted to ISO/IEC 19770-1:

Under the Primary processes:

- Business relationship management
- Service Level Management
- Security Management
- Financial Management

Under the Support processes:

- Change Management Process
- Problem Management process
- Incident Management Process
- Release Management process

For some of these processes, the word "Software" was added such as "Software release management process" to emphasize that is was targeted specifically at software management.

APPENDIX XI

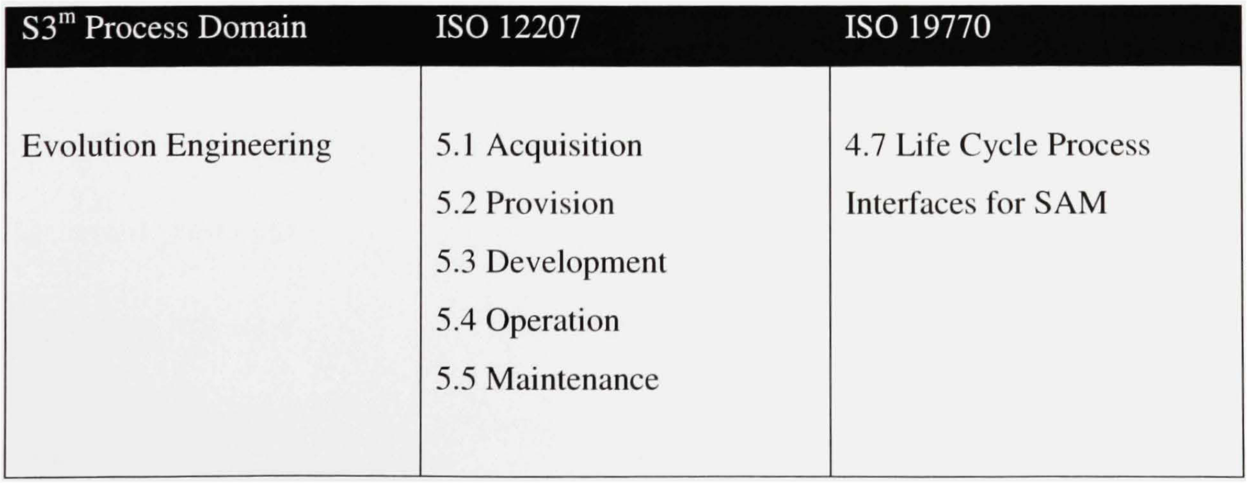
IDENTIFICATION OF PROCESS DOMAINS AND KEY PROCESS AREAS IN SAM

Table 27

Comparing categories from CMMi, S3m and ISO/IEC 19770-1

CMMi 4 Process domains	S3^{m} 4 Process Domain	SAM Process Domains
Process Management	Process Management	Process Management
Project Management	Event/Request Management	Event/Request Management?
Engineering	Evolution Engineering	Evolution Engineering?
Support	Support to Evolution Engineering	Support to Evolution Engineering?

Here ISO14764 (Maintenance) is replaced by ISO 19770 (SAM)



S3^{m} Process Domain	ISO 12207	ISO 19770
Evolution Engineering	5.1 Acquisition 5.2 Provision 5.3 Development 5.4 Operation 5.5 Maintenance	4.7 Life Cycle Process Interfaces for SAM

S3^m Process Domain	ISO 12207	ISO 19770
Support to Evolution Engineering	6.1 Documentation 6.2 Configuration Management 6.3 Quality Assurance 6.4 Verification 6.5 Validation 6.6 Joint Review 6.7 Audit 6.8 Problem Resolution	4.4 Inventory Processes for SAM 4.5 Verification and Compliance Processes for SAM 4.6 Operations Management Processes and Interfaces for SAM
Process Management / Event/Request Management	7.1 Management 7.2 Infrastructure 7.3 Improvement 7.4 Training	4.2 Control Environment for SAM 4.3 Planning and Implementation Processes for SAM

APPENDIX XII

GOALS OF INTERFACING PROCESSES

Table 28

ITIL, BS1500, ISO, SWEBOK, IEEE

Source	Grouping	Process	Goal	Activities
ISO 12207 - LIFE CYCLE	PRIMARY	MAINTENANCE	This process is activated when the software product undergoes modifications to code and associated documentation due to a problem or the need for improvement or daptation. The objective is to modify existing software product while preserving its integrity. This process includes the migration and retirement of the software product. The process ends with the retirement of the software product.	1) Process implementation;
ISO 12207 - LIFE CYCLE	PRIMARY	MAINTENANCE		2) Problem and modification analysis;
ISO 12207 - LIFE CYCLE	PRIMARY	MAINTENANCE		3) Modification implementation;
ISO 12207 - LIFE CYCLE	PRIMARY	MAINTENANCE		4) Maintenance review/acceptance
ISO 12207 - LIFE CYCLE	PRIMARY	MAINTENANCE		5) Migration;
ISO 12207 - LIFE CYCLE	PRIMARY	MAINTENANCE		6) Software retirement.
ISO 12207 - LIFE	PRIMARY	OPERATION	The process covers the	1) Process implementation;

Source	Grouping	Process	Goal	Activities
CYCLE			operation of the software product and operational support to users. Because operation of software product is integrated into the operation of the system, the activities and tasks of this process refer to the system	
ISO 12207 - LIFE CYCLE	PRIMARY	OPERATION		2) Operational testing;
ISO 12207 - LIFE CYCLE	PRIMARY	OPERATION		3) System operation;
ISO 12207 - LIFE CYCLE	PRIMARY	OPERATION		4) User support.
ISO 12207 - LIFE CYCLE	PRIMARY	DEVELOPMENT	The process contains the activities for requirements analysis, design, coding, integration, testing, and installation and acceptance related to software products. It may contain system related activities if stipulated in the contract. The developer performs or supports the activities in this process in accordance with the contract.	1) Process implementation;
ISO 12207 - LIFE CYCLE	PRIMARY	DEVELOPMENT		2) System requirements analysis;
ISO 12207 - LIFE CYCLE	PRIMARY	DEVELOPMENT		3) System architectural design;
ISO 12207 - LIFE	PRIMARY	DEVELOPMENT		4) Software requirements

Source	Grouping	Process	Goal	Activities
CYCLE				analysis;
ISO 12207 - LIFE CYCLE	PRIMARY	DEVELOPMENT		5) Software architectural design;
ISO 12207 - LIFE CYCLE	PRIMARY	DEVELOPMENT		6) Software detailed design;
ISO 12207 - LIFE CYCLE	PRIMARY	DEVELOPMENT		7) Software coding and testing;
ISO 12207 - LIFE CYCLE	PRIMARY	DEVELOPMENT		8) Software integration;
ISO 12207 - LIFE CYCLE	PRIMARY	DEVELOPMENT		9) Software qualification testing;
ISO 12207 - LIFE CYCLE	PRIMARY	DEVELOPMENT		10) System integration;
ISO 12207 - LIFE CYCLE	PRIMARY	DEVELOPMENT		11) System qualification testing
ISO 12207 - LIFE CYCLE	PRIMARY	DEVELOPMENT		12) Software installation
ISO 12207 - LIFE CYCLE	PRIMARY	DEVELOPMENT		13) Software acceptance support.
ISO 12207 - LIFE CYCLE	PRIMARY	SUPPLY	The process may be initiated either by a decision to prepare a proposal to answer an acquirer's request for proposal or by signing and entering into a contract with the acquirer to provide the system, software product or software service. The process continues with the determination of procedures and resources needed to manage and assure the project, including development of	1) Initiation;

Source	Grouping	Process	Goal	Activities
			project plans and execution of the plans through delivery of the system, software product or software service to the acquirer.	
ISO 12207 - LIFE CYCLE	PRIMARY	SUPPLY		2) Preparation of response
ISO 12207 - LIFE CYCLE	PRIMARY	SUPPLY		3) Contract;
ISO 12207 - LIFE CYCLE	PRIMARY	SUPPLY		4) Planning;
ISO 12207 - LIFE CYCLE	PRIMARY	SUPPLY		5) Execution and control;
ISO 12207 - LIFE CYCLE	PRIMARY	SUPPLY		6) Review and evaluation;
ISO 12207 - LIFE CYCLE	PRIMARY	SUPPLY		7) Delivery and completion.
ISO 12207 - LIFE CYCLE	PRIMARY	ACQUISITION	The process begins with the definition of the need to acquire a system, software product or software service. The process continues with the preparation and issue of a request for proposal, selection of a supplier, and management of the acquisition process through to the acceptance of the system, software product or software service	1) Initiation;
ISO 12207 - LIFE CYCLE	PRIMARY	ACQUISITION		2) Request-for-Proposal [-tender] preparation;
ISO 12207	PRIMARY	ACQUISITION		3) Contract

Source	Grouping	Process	Goal	Activities
- LIFE CYCLE				preparation and update;
ISO 12207 - LIFE CYCLE	PRIMARY	ACQUISITION		4) Supplier monitoring
ISO 12207 - LIFE CYCLE	PRIMARY	ACQUISITION		5) Acceptance and completion.
ISO 12207 - LIFE CYCLE	SUPPORTING	DOCUMENTATION	The Documentation Process is a process for recording information produced by a life cycle process or activity. The process contains the set of activities, which plan, design, develop, produce, edit, distribute, and maintain those documents needed by all concerned such as managers, engineers, and users of the system or software product.	1) Process implementation;
ISO 12207 - LIFE CYCLE	SUPPORTING	DOCUMENTATION		2) Design and development
ISO 12207 - LIFE CYCLE	SUPPORTING	DOCUMENTATION		3) Production;
ISO 12207 - LIFE CYCLE	SUPPORTING	DOCUMENTATION		4) Maintenance.
ISO 12207 - LIFE CYCLE	SUPPORTING	CONFIGURATION MANAGEMENT	The Configuration Management Process is a process of applying administrative and technical procedures throughout the software life cycle to: identify, define, and baseline	1) Process implementation;

Source	Grouping	Process	Goal	Activities
			software items in a system; control modifications and releases of the items; record and report the status of the items and modification requests; ensure the completeness, consistency, and correctness of the items; and control storage, handling, and delivery of the items.	
ISO 12207 - LIFE CYCLE	SUPPORTING	CONFIGURATION MANAGEMENT		2) Configuration identification;
ISO 12207 - LIFE CYCLE	SUPPORTING	CONFIGURATION MANAGEMENT		3) Configuration control;
ISO 12207 - LIFE CYCLE	SUPPORTING	CONFIGURATION MANAGEMENT		4) Configuration status accounting;
ISO 12207 - LIFE CYCLE	SUPPORTING	CONFIGURATION MANAGEMENT		5) Configuration evaluation;
ISO 12207 - LIFE CYCLE	SUPPORTING	CONFIGURATION MANAGEMENT		6) Release management and delivery.
ISO 12207 - LIFE CYCLE	SUPPORTING	QUALITY ASSURANCE	The Quality Assurance Process is a process for providing adequate assurance that the software products and processes in the project life cycle conform to their specified requirements and adhere to their established plans.	1) Process implementation;
ISO 12207 - LIFE CYCLE	SUPPORTING	QUALITY ASSURANCE		2) Product assurance
ISO 12207 - LIFE CYCLE	SUPPORTING	QUALITY ASSURANCE		3) Process assurance

Source	Grouping	Process	Goal	Activities
ISO 12207 - LIFE CYCLE	SUPPORTING	QUALITY ASSURANCE		4) Assurance of quality systems.
ISO 12207 - LIFE CYCLE	SUPPORTING	VERIFICATION	The Verification Process is a process for determining whether the software products of an activity fulfill the requirements or conditions imposed on them in the previous activities. For cost and performance effectiveness, verification should be integrated, as early as possible, with the process (such as supply, development, operation, or maintenance) that employs it. This process may include analysis, review and test.	1) Process implementation;
ISO 12207 - LIFE CYCLE	SUPPORTING	VERIFICATION		2) Verification
ISO 12207 - LIFE CYCLE	SUPPORTING	VALIDATION	The Validation Process is a process for determining whether the requirements and the final, as-built system or software product fulfills its specific intended use. Validation may be conducted in earlier stages. This process may be conducted as a part of Software Acceptance Support	1) Process implementation
ISO 12207	SUPPORTING	VALIDATION		2) Validation.

Source	Grouping	Process	Goal	Activities
- LIFE CYCLE				
ISO 12207 - LIFE CYCLE	SUPPORTING	JOINT REVIEW	The Joint Review Process is a process for evaluating the status and products of an activity of a project as appropriate. Joint reviews are at both project management and technical levels and are held throughout the life of the contract.	1) Process implementation;
ISO 12207 - LIFE CYCLE	SUPPORTING	JOINT REVIEW		2) Project management reviews;
ISO 12207 - LIFE CYCLE	SUPPORTING	JOINT REVIEW		3) Technical reviews.
ISO 12207 - LIFE CYCLE	SUPPORTING	AUDIT	The Audit Process is a process for determining compliance with the requirements, plans, and contract as appropriate.	1) Process implementation
ISO 12207 - LIFE CYCLE	SUPPORTING	AUDIT		2) Audit.
ISO 12207 - LIFE CYCLE	SUPPORTING	PROBLEM RESOLUTION	The Problem Resolution Process is a process for analyzing and resolving the problems (including nonconformances), whatever their nature or source, that are discovered during the execution of development, operation, maintenance, or other processes.	1) Process implementation;

Source	Grouping	Process	Goal	Activities
ISO 12207 - LIFE CYCLE	SUPPORTING	PROBLEM RESOLUTION		2) Problem resolution
ISO 12207 - LIFE CYCLE	ORGANIZATIONAL	MANAGEMENT	The Management Process contains the generic activities and tasks, which may be employed by any party that has to manage its respective process(es). The manager is responsible for product management, project management, and task management of the applicable process(es), such as the acquisition, supply, development, operation, maintenance, or supporting process.	1) Initiation and scope definition;
ISO 12207 - LIFE CYCLE	ORGANIZATIONAL	MANAGEMENT		2) Planning;
ISO 12207 - LIFE CYCLE	ORGANIZATIONAL	MANAGEMENT		3) Execution and control;
ISO 12207 - LIFE CYCLE	ORGANIZATIONAL	MANAGEMENT		4) Review and evaluation;
ISO 12207 - LIFE CYCLE	ORGANIZATIONAL	MANAGEMENT		5) Closure
ISO 12207 - LIFE CYCLE	ORGANIZATIONAL	INFRASTRUCTURE	The Infrastructure Process is a process to establish and maintain the infrastructure needed for any other process. The infrastructure may include hardware,	1) Process implementation;

Source	Grouping	Process	Goal	Activities
			software, tools, techniques, standards, and facilities for development, operation, or maintenance.	
ISO 12207 - LIFE CYCLE	ORGANIZATIONAL	INFRASTRUCTURE		2) Establishment of the infrastructure;
ISO 12207 - LIFE CYCLE	ORGANIZATIONAL	INFRASTRUCTURE		3) Maintenance of the infrastructure.
ISO 12207 - LIFE CYCLE	ORGANIZATIONAL	IMPROVEMENT	The Improvement Process is a process for establishing, assessing, measuring, controlling, and improving a software life cycle process.	1) Process establishment;
ISO 12207 - LIFE CYCLE	ORGANIZATIONAL	IMPROVEMENT		2) Process assessment;
ISO 12207 - LIFE CYCLE	ORGANIZATIONAL	IMPROVEMENT		3) Process improvement.
ISO 12207 - LIFE CYCLE	ORGANIZATIONAL	TRAINING	The Training Process is a process for providing and maintaining trained personnel. The acquisition, supply, development, operation, or maintenance of software products is largely dependent upon knowledgeable and skilled personnel.	1) Process implementation;
ISO 12207 - LIFE CYCLE	ORGANIZATIONAL	TRAINING		2) Training material development;
ISO 12207 - LIFE CYCLE	ORGANIZATIONAL	TRAINING		3) Training plan implementation.

Source	Grouping	Process	Goal	Activities
SWEBOK		SOFTWARE REQUIREMENTS	The Software Requirements Knowledge Area (KA) is concerned with the elicitation, analysis, specification, and validation of software requirements.	1- Software Requirements Fundamentals
SWEBOK		SOFTWARE REQUIREMENTS		**2- Requirements Process**
SWEBOK		SOFTWARE REQUIREMENTS		**3- Requirements Elicitation**
SWEBOK		SOFTWARE REQUIREMENTS		**4- Requirements Analysis**
SWEBOK		SOFTWARE REQUIREMENTS		**5- Requirements Specification**
SWEBOK		SOFTWARE REQUIREMENTS		**6- Requirements validation**
SWEBOK		SOFTWARE REQUIREMENTS		**7- Practical Considerations**
SWEBOK		SOFTWARE DESIGN	Design is defined in as both "the process of defining the architecture, components, interfaces, and other characteristics of a system or component" and "the result of [that] process."	1- **Software Design Fundamentals**
SWEBOK		SOFTWARE DESIGN		2- **Key Issues in Software Design**
SWEBOK		SOFTWARE DESIGN		**3- Software Structure and Architecture**
SWEBOK		SOFTWARE CONSTRUCTION	The term software construction refers to the detailed creation of working, meaningful software through a combination of coding, verification, unit testing, integration testing, and debugging	1- **Software Construction Fundamentals**

Source	Grouping	Process	Goal	Activities
SWEBOK		SOFTWARE CONSTRUCTION		2- **Managing Construction**
SWEBOK		SOFTWARE CONSTRUCTION		**3- Practical considerations**
SWEBOK		SOFTWARE TESTING	Testing is an activity performed for evaluating product quality, and for improving it, by identifying defects and problems. Software testing consists of the dynamic verification of the behavior of a program on a finite set of test cases, suitably selected from the usually infinite	1- Software Testing Fundamentals
SWEBOK		SOFTWARE TESTING		2- **Test Levels**
SWEBOK		SOFTWARE TESTING		**3- Test Techniques**
SWEBOK		SOFTWARE TESTING		**4- Test-related measures**
SWEBOK		SOFTWARE TESTING		**5- Test Process**
SWEBOK		SOFTWARE MAINTENANCE	Software development efforts result in the delivery of a software product which satisfies user requirements. Accordingly, the software product must change or evolve. Once in operation, defects are uncovered, operating environments change, and new user requirements surface. The maintenance phase of the life	1- **Software Maintenance Fundamentals**

Source	Grouping	Process	Goal	Activities
			cycle begins following a warranty period or post-implementation support delivery, but maintenance activities occur much earlier.	
SWEBOK		SOFTWARE MAINTENANCE		**2- Key Issues in Software Maintenance**
SWEBOK		SOFTWARE MAINTENANCE		**3- Maintenance Process**
SWEBOK		SOFTWARE MAINTENANCE		**4- Techniques for Maintenance**
SWEBOK		SOFTWARE CONFIGURATION MANAGEMENT	A system can be defined as a collection of components organized to accomplish a specific function or set of functions. The configuration of a system is the functional and/or physical characteristics of hardware, firmware, or software, or a combination of these, as set forth in technical documentation and achieved in a product.	**1- Management of the SCM Process**
SWEBOK		SOFTWARE CONFIGURATION MANAGEMENT		**2- Software Configuration Identification**
SWEBOK		SOFTWARE CONFIGURATION MANAGEMENT		**3- Software Configuration Control**
SWEBOK		SOFTWARE CONFIGURATION MANAGEMENT		**4- Software Configuration Status Accounting**
SWEBOK		SOFTWARE CONFIGURATION MANAGEMENT		**5- Software Configuration Auditing**
SWEBOK		SOFTWARE CONFIGURATION		**6- Software**

Source	Grouping	Process	Goal	Activities
		MANAGEMENT		**Release Management and Delivery**
SWEBOK		SOFTWARE ENGINEERING MANAGEMENT	Software Engineering Management can be defined as the application of management activities—planning, coordinating, measuring, monitoring, controlling, and reporting—to ensure that the development and maintenance of software is systematic, disciplined, and quantified	**1- Initiation and Scope Definition**
SWEBOK		SOFTWARE ENGINEERING MANAGEMENT		**2- Software Project Planning**
SWEBOK		SOFTWARE ENGINEERING MANAGEMENT		**3- Software Project Enactment**
SWEBOK		SOFTWARE ENGINEERING MANAGEMENT		**4- Review and Evaluation**
SWEBOK		SOFTWARE ENGINEERING MANAGEMENT		**5- Closure**
SWEBOK		SOFTWARE ENGINEERING MANAGEMENT		**6- Software Engineering Measurement**
SWEBOK		SOFTWARE ENGINEERING PROCESS	The Software Engineering Process KA can be examined on two levels. The first level encompasses the technical and managerial activities within the software life cycle processes that are performed during software acquisition, development, maintenance, and	**1- Process Implementation and Change**

Source	Grouping	Process	Goal	Activities
			retirement. The second is the meta-level, which is concerned with the definition, implementation, assessment, measurement, management, change, and improvement of the software life cycle processes themselves. The first level is covered by the other KAs in the Guide. This KA is concerned with the second.	
SWEBOK		SOFTWARE ENGINEERING PROCESS		**2- Process Definition**
SWEBOK		SOFTWARE ENGINEERING PROCESS		**3- Process Assessment**
SWEBOK		SOFTWARE ENGINEERING PROCESS		**4- Process and Product Measurement**
SWEBOK		SOFTWARE ENGINEERING TOOLS AND METHODS	Software development tools are the computer-based tools that are intended to assist the software life cycle processes.Tools allow repetitive, well-defined actions to be automated, reducing the cognitive load on the software engineer who is then free to concentrate on the creative aspects of the process. Tools are often designed to support particular software engineering	**1- Software Engineering Tools**

Source	Grouping	Process	Goal	Activities
			methods, reducing any administrative load associated with applying the method manually.	
SWEBOK		SOFTWARE ENGINEERING TOOLS AND METHODS		2- **Software Engineering Methods**
SWEBOK		SOFTWARE QUALITY	This chapter deals with software quality considerations which transcend the life cycle processes. Software quality is a ubiquitous concern in software engineering, and so it is also considered in many of the Kas	1- **Software Quality Fundamentals**
SWEBOK		SOFTWARE QUALITY		2- **Software Quality Management Processes**
SWEBOK		SOFTWARE QUALITY		**3- Practical Considerations**
ITIL	Service support	Incident Management	The primary goal of the Incident Management process is to restore normal service operation as quickly as possible and minimize the adverse impact on business operations	Incident detection and recording
ITIL	Service support	Incident Management		Classification and initial support
ITIL	Service support	Incident Management		Classification and initial support
ITIL	Service support	Incident Management		Resolution and recovery

Source	Grouping	Process	Goal	Activities
ITIL	Service support	Incident Management		Incident closure
ITIL	Service support	Incident Management		Ownership, monitoring, tracking and communication
ITIL	Service support	Problem Management	The goal of Problem Management is to minimize the adverse impact of Incidents and Problems on the business that are caused by errors within the IT Infrastructure, and to prevent recurrence of Incidents related to these errors	Problem identification and recording
ITIL	Service support	Problem Management		Problem classification
ITIL	Service support	Problem Management		Problem investigation and diagnosis
ITIL	Service support	Problem Management		Error identification and recording
ITIL	Service support	Problem Management		Error assessment
ITIL	Service support	Problem Management		Error resolution recording
ITIL	Service support	Problem Management		Error closure
ITIL	Service support	Problem Management		Problem/error resolution monitoring
ITIL	Service support	Configuration Management	Account for all the IT assets and configurations within the organization and its services provide accurate information on configurations and their	Configuration Management planning

Source	Grouping	Process	Goal	Activities
			documentation to support all the other Service Management processes provide a sound basis for Incident Management, Problem Management, Change Management and Release Management verify the configuration records against the infrastructure and correct any exceptions.	
ITIL	Service support	Configuration Management		Configuration identification
ITIL	Service support	Configuration Management		Control of Cis
ITIL	Service support	Configuration Management		Configuration status accounting
ITIL	Service support	Configuration Management		Configuration verification and audit
ITIL	Service support	Configuration Management		CMDB back-ups, archives and housekeeping
ITIL	Service support	Change Management	The goal of the Change Management process is to ensure that standardized methods and procedures are used for efficient and prompt handling of all Changes, in order to minimize the impact of Change-related Incidents upon service	Planning the implementation of operational processes

Source	Grouping	Process	Goal	Activities
			quality, and consequently to improve the day-to-day operations of the organization.	
ITIL	Service support	Change Management		Change logging and filtering
ITIL	Service support	Change Management		Allocation of priorities
ITIL	Service support	Change Management		Change categorization
ITIL	Service support	Change Management		CAB meetings
ITIL	Service support	Change Management		Impact and resource assessment
ITIL	Service support	Change Management		Change approval
ITIL	Service support	Change Management		Change scheduling
ITIL	Service support	Change Management		Change building, testing and implementation
ITIL	Service support	Change Management		Urgent Changes
ITIL	Service support	Change Management		Urgent Change building, testing and implementation
ITIL	Service support	Change Management		Change review
ITIL	Service support	Change Management		Reviewing the Change Management process for efficiency and effectiveness
ITIL	Service support	Change Management		Roles and responsibilities
ITIL	Service support	Release Management	To plan and oversee the successful rollout of software and related hardware	Release planning
ITIL	Service support	Release Management		Designing, building and configuring a Release

Source	Grouping	Process	Goal	Activities
ITIL	Service support	Release Management		Release acceptance
ITIL	Service support	Release Management		Rollout planning
ITIL	Service support	Release Management		Communication, preparation and training
ITIL	Service support	Release Management		Distribution and installation
ITIL	Service Delivery	Service Level Management	The goal for SLM is to maintain and improve IT Service quality, through a constant cycle of agreeing, monitoring and reporting upon IT Service achievements and instigation of actions to eradicate poor service - in line with business or Cost justification	Initial planning activities
ITIL	Service Delivery	Service Level Management		Plan monitoring capabilities
ITIL	Service Delivery	Service Level Management		Establish initial perception of the Services
ITIL	Service Delivery	Service Level Management		Underpinning contracts and Operational Level Agreements
ITIL	Service Delivery	Service Level Management		Produce a Service Catalogue
ITIL	Service Delivery	Service Level Management		Expectation Management
ITIL	Service Delivery	Service Level Management		Plan the SLA structure
ITIL	Service Delivery	Service Level Management		Establish Service Level Requirements

Source	Grouping	Process	Goal	Activities
				and Draft SLA
ITIL	Service Delivery	Service Level Management		Wording of SLAs
ITIL	Service Delivery	Service Level Management		Seek agreement
ITIL	Service Delivery	Service Level Management		Establish monitoring capabilities
ITIL	Service Delivery	Service Level Management		Review Underpinning contracts and Operational Level Agreements
ITIL	Service Delivery	Service Level Management		Define Reporting and Review Procedures
ITIL	Service Delivery	Service Level Management		Publicise the existence of SLAs
ITIL	Service Delivery	Service Level Management		Monitoring and Reporting
ITIL	Service Delivery	Service Level Management		Service review meetings
ITIL	Service Delivery	Service Level Management		Service Improvement Programme
ITIL	Service Delivery	Service Level Management		Maintenance of SLAs, contracts and OLAs
ITIL	Service Delivery	Financial Management	To be able to account fully for the spend on IT Services and to attribute these costs to the services delivered to the organisation's Customers. To assist management decisions on IT investment by	Developing the IT Accounting System

Source	Grouping	Process	Goal	Activities
			providing detailed business cases for Changes to IT Services.	
ITIL	Service Delivery	Financial Management		Developing the Charging System
ITIL	Service Delivery	Financial Management		Planning for IT Accounting and Charging
ITIL	Service Delivery	Financial Management		Implementation
ITIL	Service Delivery	Financial Management		Ongoing management and operation
ITIL	Service Delivery	IT Service Continuity Management	To support the overall Business Continuity Management process by ensuring that the required IT technical and services facilities can be recovered within required, and agreed, business timescales.	Scope of ITSCM
ITIL	Service Delivery	IT Service Continuity Management		The Business Continuity Lifecycle
ITIL	Service Delivery	IT Service Continuity Management		Management Structure
ITIL	Service Delivery	IT Service Continuity Management		Generating awareness
ITIL	Service Delivery	Capacity Management	Capacity Management needs to understand the business requirements (the required Service Delivery), the organization's operation (the current Service Delivery) and the	Business Capacity Management

Source	Grouping	Process	Goal	Activities
			IT Infrastructure (the means of Service Delivery), and ensure that all the current and future Capacity and performance aspects of the business requirements are provided cost-effectively.	
ITIL	Service Delivery	Capacity Management		Service Capacity Management
ITIL	Service Delivery	Capacity Management		Resource Capacity Management
ITIL	Service Delivery	Capacity Management		Monitoring
ITIL	Service Delivery	Capacity Management		Analysis
ITIL	Service Delivery	Capacity Management		Tuning
ITIL	Service Delivery	Capacity Management		Implementation
ITIL	Service Delivery	Capacity Management		Storage of Capacity Management data
ITIL	Service Delivery	Capacity Management		Demand Management
ITIL	Service Delivery	Capacity Management		Modelling
ITIL	Service Delivery	Capacity Management		Application sizing
ITIL	Service Delivery	Capacity Management		Production of the Capacity Plan
ITIL	Service Delivery	Availability Management	To optimize the capability of the IT Infrastructure, services and supporting organisation to deliver a Cost	Availability Planning

Source	Grouping	Process	Goal	Activities
			effective and sustained level of Availability that enables the business to satisfy its business objectives	
ITIL	Service Delivery	Availability Management		Availability improvement
ITIL	Service Delivery	Availability Management		Availability measurement and reporting
ITIL	Service Delivery	Availability Management		Availability Management tools
ITIL	Service Delivery	Availability Management		Availability Management methods and techniques
IIMM		Corporate AM Direction		Identify corporate need
IIMM		Corporate AM Direction		Obtain organisational commitment
IIMM		Corporate AM Direction		Adopt corporate AM goals and objectives
IIMM		Corporate AM Direction		Define AM roles and responsabilities
IIMM		Corporate AM Direction		Involve key stakeholders
IIMM		The AM Team		Oversee AM Implementation
IIMM		The AM Team		Coordinate AM activities
IIMM		The AM Team		Internal audit
IIMM		Asset Management Improvement Planning		AM Status Review
IIMM		Asset Management Improvement Planning		The improvement programme
IIMM		Asset Management Improvement Planning		Pilot Studies
IIMM		The AM Plan		Prepare AM

Source	Grouping	Process	Goal	Activities
				Plan
IIMM		The AM Plan		Develop lifecycle strategies
IIMM		Implementing the AMP		Information flows
IIMM		Implementing the AMP		Service Delivery issues
IIMM		AMP review and audit		AM performance
IIMM		AMP review and audit		Techinical content of AMP
IIMM		AMP review and audit		Compliance with legal requirements
IIMM		AMP review and audit		Internal/external audits
ISO 19770-1	Primary Processes	Inventory Processes for SAM		Software Asset Identification
ISO 19770-1	Primary Processes	Inventory Processes for SAM		Software Asset Inventory Management
ISO 19770-1	Primary Processes	Inventory Processes for SAM		Software Asset Control
ISO 19770-1	Primary Processes	Verification and Compliance Processes for SAM		Software Asset Record Verification
ISO 19770-1	Primary Processes	Verification and Compliance Processes for SAM		Software Asset Licensing compliance
ISO 19770-1	Primary Processes	Verification and Compliance Processes for SAM		Software Asset Security compliance
ISO 19770-1	Primary Processes	Verification and Compliance Processes for SAM		Conformance verification for SAM
ISO 19770-1	Primary Processes	Operations Management Processes and Interfaces for SAM		Relationship and contract management for SAM
ISO 19770-1	Primary Processes	Operations Management Processes and Interfaces for SAM		Financial Management for SAM
ISO 19770-1	Primary Processes	Operations Management Processes and Interfaces for SAM		Service Level Management for SAM
ISO	Primary Processes	Operations Management Processes and		Security

Source	Grouping	Process	Goal	Activities
19770-1		Interfaces for SAM		Management for SAM
ISO 19770-1	Support Processes	Life Cycle Interfaces for SAM		Change Management Process
ISO 19770-1	Support Processes	Life Cycle Interfaces for SAM		Software development process
ISO 19770-1	Support Processes	Life Cycle Interfaces for SAM		Software Deployment process
ISO 19770-1	Support Processes	Life Cycle Interfaces for SAM		Problem Management process
ISO 19770-1	Support Processes	Life Cycle Interfaces for SAM		Acquisition Process
ISO 19770-1	Support Processes	Life Cycle Interfaces for SAM		Software Release Management process
ISO 19770-1	Support Processes	Life Cycle Interfaces for SAM		Incident Management Process
ISO 19770-1	Support Processes	Life Cycle Interfaces for SAM		Retirement Process
ISO 19770-1	Organisational Processes	Control Environement for SAM		Corporate Governance
ISO 19770-1	Organisational Processes	Control Environement for SAM		Roles and responsibility
ISO 19770-1	Organisational Processes	Control Environement for SAM		Policies processes and procedures
ISO 19770-1	Organisational Processes	Control Environement for SAM		Competence in SAM
ISO 19770-1	Organisational Processes	Planning and Implementation Processes for SAM		Planning for SAM
ISO 19770-1	Organisational Processes	Planning and Implementation Processes for SAM		Implementation for SAM
ISO 19770-1	Organisational Processes	Planning and Implementation Processes for SAM		Monitoring and review for SAM
ISO 19770-1	Organisational Processes	Planning and Implementation Processes for SAM		Continual improvement for SAM
ISO 15939		Establish and sustain measurement commitment		Accept the requirements for measurement
ISO 15939		Establish and sustain measurement commitment		Assign resources
ISO		Plan the measurement process		Characterise

Source	Grouping	Process	Goal	Activities
15939				organisational unit
ISO 15939		Plan the measurement process		Identify information needs
ISO 15939		Plan the measurement process		Select measures
ISO 15939		Plan the measurement process		Define data collection, analysis, and reporting procedures
ISO 15939		Plan the measurement process		Define criteria for evaluating the information products and the measurement process
ISO 15939		Plan the measurement process		Review, approve, and provide resources for measurement tasks
ISO 15939		Plan the measurement process		Acquire and deploy supporting technologies
ISO 15939		Perform the measurement process		Integrate procedures
ISO 15939		Perform the measurement process		Collect data
ISO 15939		Perform the measurement process		Analyse data and develop information products
ISO 15939		Perform the measurement process		Communicate results
ISO 15939		Evaluate measurement		Evaluate information products and the measurement process
ISO 15939		Evaluate measurement		Identify potential improvements
IEEE Std 1062		Software acquisition process	The software acquisition life cycle represents the period of time that begins with the decision to acquire a software product and ends	Planning organizational strategy

Source	Grouping	Process	Goal	Activities
			when the product is no longer available for use. It typically includes a planning phase, contracting phase, product implementation phase, product acceptance phase, and follow-on phase. This life cycle provides an overall framework within which most software acquisitions occur	
IEEE Std 1062		Software acquisition process		Implementing organization's process
IEEE Std 1062		Software acquisition process		Defining the software requirements
IEEE Std 1062		Software acquisition process		Identifying potential suppliers
IEEE Std 1062		Software acquisition process		Preparing contract requirements
IEEE Std 1062		Software acquisition process		Evaluating proposals and selecting supplier
IEEE Std 1062		Software acquisition process		Managing for supplier performance
IEEE Std 1062		Software acquisition process		Accepting the software
IEEE Std 1062		Software acquisition process		Using the software
BS15000	Service delivery process	Service level management	To define, agree, record and manage levels of service	
BS15000	Service delivery process	Service reporting	To produce agreed, timely, reliable, accurate reports for informed decision making and effective communication.	a) performance against service level targets;
BS15000	Service delivery process	Service reporting		b) non-compliance and issues, e.g. against the SLA,

Source	Grouping	Process	Goal	Activities
				security breech;
BS15000	Service delivery process	Service reporting		c) workload characteristics, e.g. volume, resource utilisation;
BS15000	Service delivery process	Service reporting		d) performance reporting following major events, e.g. major incidents and changes;
BS15000	Service delivery process	Service reporting		e) trend information;
BS15000	Service delivery process	Service reporting		f) satisfaction analysis.
BS15000	Service delivery process	Service continuity and availability management	To ensure that agreed service continuity and availability to customers can be met in all circumstances	
BS15000	Service delivery process	Budgeting and accounting for IT services	To budget and account for the cost of service provision	a) budgeting, and accounting for all components including IT assets, shared resources,
BS15000	Service delivery process	Budgeting and accounting for IT services		overheads, externally supplied service, people, insurance and licences;
BS15000	Service delivery process	Budgeting and accounting for IT services		b) apportioning indirect costs and allocating direct costs to services;
BS15000	Service delivery process	Budgeting and accounting for IT services		c) effective financial control and authorization.
BS15000	Service delivery process	Information security management	To manage information security effectively within all service activities	a) implement the requirements of the information security policy;
BS15000	Service delivery process	Information security management		b) manage risks associated with access to the service or systems.
BS15000	Relationship processes	Business relationship management	To establish and maintain a good relationship between the service provider and the customer based on understanding the customer and their business drivers	

Source	Grouping	Process	Goal	Activities
BS15000	Relationship processes	Supplier management		
BS15000	Resolution processes	Incident management	To restore agreed service to the business as soon as possible or to respond to service requests.	
BS15000	Resolution processes	Problem management	To minimize disruption to the business by proactive identification and analysis of the cause of service incidents and by managing problems to closure	
BS15000	Control processes	Configuration management	To define and control the components of the service and infrastructure and maintain accurate configuration information	
BS15000	Control processes	Change management	To ensure all changes are assessed, approved, implemented and reviewed in a controlled manner.	
BS15000	Release process	Release management process	To deliver, distribute and track one or more changes in a release into the live environment.	

APPENDIX XIII

MEETING MINUTES, SC7 WG21, SAM, MONTREAL, CANADA, 2003-05-12,16 (N2851)

Comments and descriptions for changes made in the working document for ISO/IEC 19770-1 Software Asset Management Process and 19770-2 Software Asset Management Tag

1. (19770-1) Text describing the purpose of the standard was moved from the introduction part to section 1.1 Purpose and the wording was slightly changed.
2. (19770-1) Section 1.2 – Field of application- have been slightly changed, the bullet form has been taken away to suite into the ISO template and form to write international standards. Following text has also been added: This International Standard applies to the software asset management process for software assets in form of acquired or self-made software applications
3. Comments from USA:

Involve the legal departments in companies and organizations and make then aware of all aspects of software license administration.

Inform and try to involve the mayor software houses.

-3-

Make the American software purchasing management and other equal managers aware of the fact that in Europe standards are looked upon almost as laws.

Add some text about license types in the standard (as we had in the strawman).

4. Comments from Canada:

(19770-1) Have you considered including Configuration Management in the standard?

(19770-2) Have you considered what to do when there is a mismatch between the information in the tag and the application?

(19770-2) Network devices can identify themselves. Can you do it in the same way?

A tag itself cannot solve everything. A process needs to be in place as well.

(19770-2) Dictionaries can have unique markers.

Cannot a supplier demand a client to use the standard?

Take away the automatic update via Internet for SW applications.

(19770-2) Have you considered how the maintain the tag standard? The variables might have to be updated or new variables might have to be added. How will that be managed?

5. Comments from Germany:

The standard can be used by the supplier in such a way that the supplier will not sell the application if the customers do not implement the standard.

(19770-2) Have you considered using bar code as a tag?

(19770-2) Hook into any existing coding standard (regarding the tag)

(19770-2) Will WG21 provides a tool for suppliers and clients for producing and implementing the tag?

6. Comments from Sweden:

(19770-2) Investigate the possibility to use the Common Information Model (CIM) as the source for the tag. (More info can be found at www.dmtf.org)

(19770-2) Have you considered using a so-called "mother tag" for the applications defining: 'these files belong to this application'?

(19770-1) Consider adding explanations about the roles and responsibilities for Global and Local SAM organizations.

APPENDIX XIV

MEETING MINUTES –JTC 1/SC 7/WG 21 MEETING IN STOCKHOLM, SWEDEN, NOVEMBER 3-6, 2003 (N2962)

3.5.1 19770-1 Revision Status

A WD was produced after the plenary meeting in Montréal and a combined WD, CD Registration and CD ballot was distributed. The comments from the ballot have been incorporated into the document according to the following table:

ISO/IEC JTC 1/SC 7 N

The letter ballot summary (doc. SC 7 N 2934) and the comments received, indicates that a second version of the WD shall be issued.

Action on WG21 as a group: Issue a WD2 for WG review or combined WD, CD Registration and CD ballot January 5 2004.

Action on David Déry: Make a diagram describing the software asset management process. Due date January 5th 2004.

3.5.2 19770-2 Revision Status

During the meeting XML was considered as a well suitable format for the software id tag. However, XML can be implemented at several complex levels and it was decided that three different implementation levels should be reviewed by WG21. After the 30 days review the level of implementation will be decided depending on the comments. An expert team from AstraZeneca has developed the three different implementation levels as well as recommending one of the levels.

The group does not want to tie the tag to XML but exemplifying the tag with XML will be a good guidance for organizations developing their own software identification tags.

19770-2 will be sent out together with 19770-1.

Action on WG21 as a group: Issue a WD2 for WG review

APPENDIX XV

CANADIAN ISO/IEC 19770-1 COMMENTS– APRIL 2006

FDIS ISO/IEC 19770-1 Standard on Software Asset Management

Comments from Canada:

1. In section 1.2: 2nd paragraph of 1st note

   Comment: Including non-executables in the scope may generate a difference in scope with 19770-2: the TAG may be used to limit the execution of certain software: this is possible only for executable software

   Recommendation: A note may be necessary in 19770-2 to warn the reader of the possible difference in scope between the processes and the TAG.

2. In section 2.1: 3rd line of 1st paragraph

   Comment: Making a reference to outsourced process without context makes it difficult to understanding why it is there.

   Recommendation: If the wording is to remain as it is, a note may be necessary stating that all business processes of the organization must be included: even those performed by an outsourced on behalf of the organization being assessed.

3. In section 3.13

   Comment: Is the definition "software header" necessary as it is not really used by the standard? (Only within the definitions).

4. In section 4.1.1

   Comment: Has terms and definitions of ISO/IEC 20000 been reviewed to ensure that there is no conflict since alignment with this standard is stated or it is just a high level alignment?

Recommendation: The term "closely aligned" could be changed to just "aligned"; or a note may be required to state that any differences are not indented.

5. In section 4.1.2

Comment: ISO/IEC 12207 also divides its processes in 3 categories: Primary life cycle processes, supporting process and organizational life cycle processes.

Recommendation: It might be strategic to align with 12207 or otherwise make a note explaining the difference and/or mapping between the 2 sets of category labels

6. In section 4.3.5

Comment: In the Continual improvements section for SAM, it is recommended that data/measures be collected. However, it is not said that they must be used. It might be beyond the scope of this standard to say that a measurement program must be in place.

Recommendation: It is not recommended to force organizations to have a measurement program to properly use the information collected, however a note could be added to refer to ISO 15939 – "Software measurement process" for guidance on how to implement an organizational measurement program.

APPENDIX XVI

WG MEETING MINUTES SC7/WG21 - SOFTWARE ASSET MANAGEMENT - BRISBANE, AUSTRALIA MAY 10TH – 14TH, 2004 (N3088)

Comments and descriptions for changes made in the working document for ISO/IEC 19770-1 Software Asset Management Process and 19770-2 Software Asset Management Tag:

1. 19770-2 should only be a logical and not technical standard

2. Retirement of software has to be better described; suggestion is to have it detailed as other processes.

3. Term 'Organization Chart' to be better explained.

4. Change wording for Roles and Responsibilities.

5. The terms of 'global' with respect to 'local' was discussed and changed.

6. Structure of description of processes is not uniformed

7. Code of Conduct could be Development process, communication process and management process. How would this come in line with the other processes?

8. Check if process maintenance renewal is correct.

9. Check if the term 'establish' is defined in any ISO-SC7

10. All bullet point to be replaced with numbered point; alphabetic or numeric.

12. Comment from Finland: does software include a full system or only the software components.

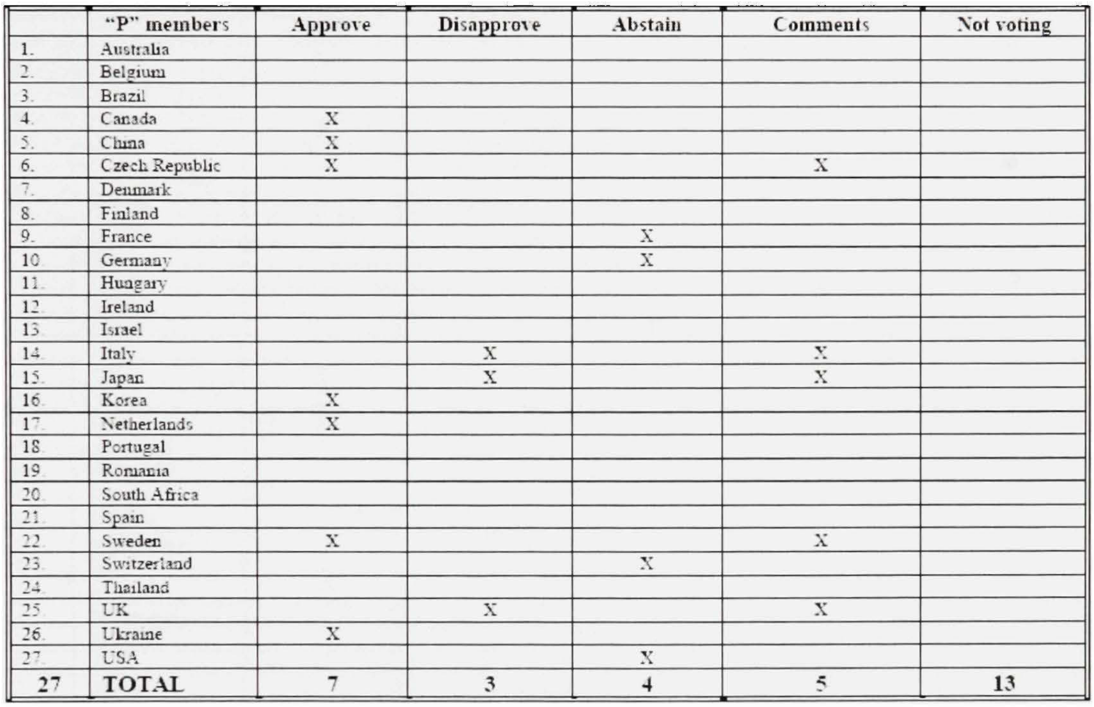
13. Comment WG2: check for need for software package documentation.

All comments that were disposed of during the meeting have been recorded and described in the document entitled: "Changes Registry, WG21, Plenary 2004".

APPENDIX XVII

LETTER BALLOT SUMMARY - N3084 - CD19770.2 – INFORMATION TECHNOLOGY– SOFTWARE ASSET MANAGEMENT PROCESS (N3125)

TABLE OF VOTING AND COMMENTS RECEIVED

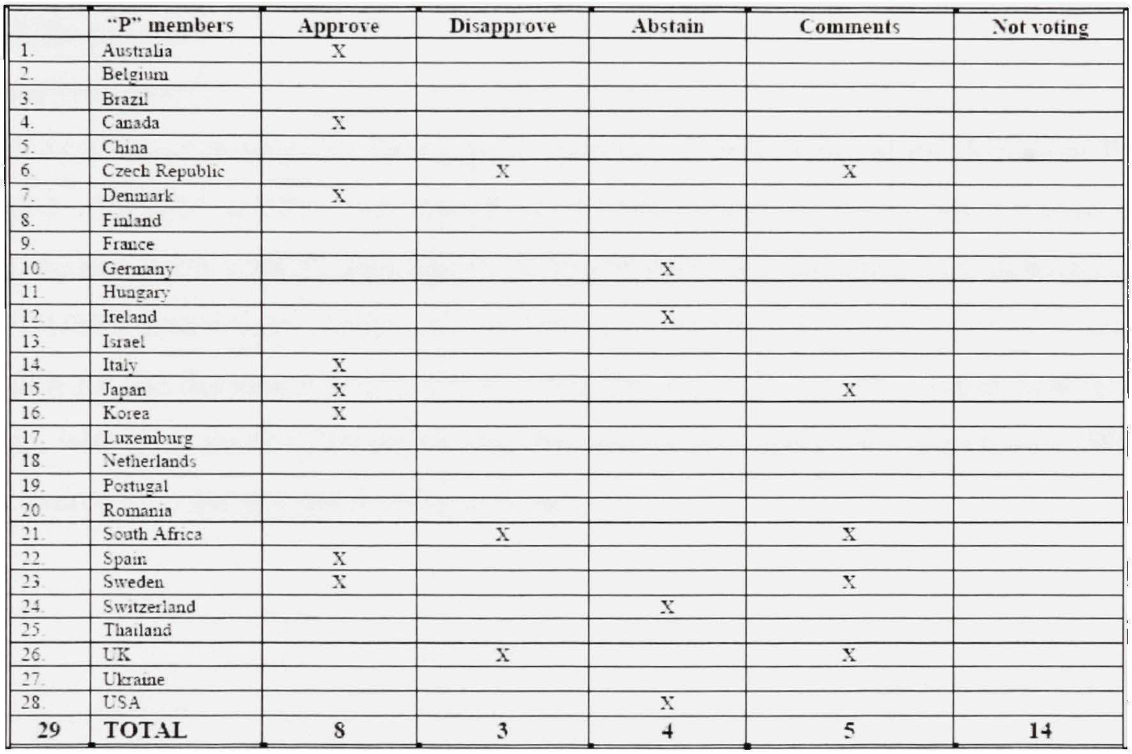


	"P" members	Approve	Disapprove	Abstain	Comments	Not voting
1.	Australia					
2.	Belgium					
3.	Brazil					
4.	Canada	X				
5.	China	X				
6.	Czech Republic	X			X	
7.	Denmark					
8.	Finland					
9.	France			X		
10.	Germany			X		
11.	Hungary					
12.	Ireland					
13.	Israel					
14.	Italy		X		X	
15.	Japan		X		X	
16.	Korea	X				
17.	Netherlands	X				
18.	Portugal					
19.	Romania					
20.	South Africa					
21.	Spain					
22.	Sweden	X			X	
23.	Switzerland			X		
24.	Thailand					
25.	UK		X		X	
26.	Ukraine	X				
27.	USA			X		
27	TOTAL	7	3	4	5	13

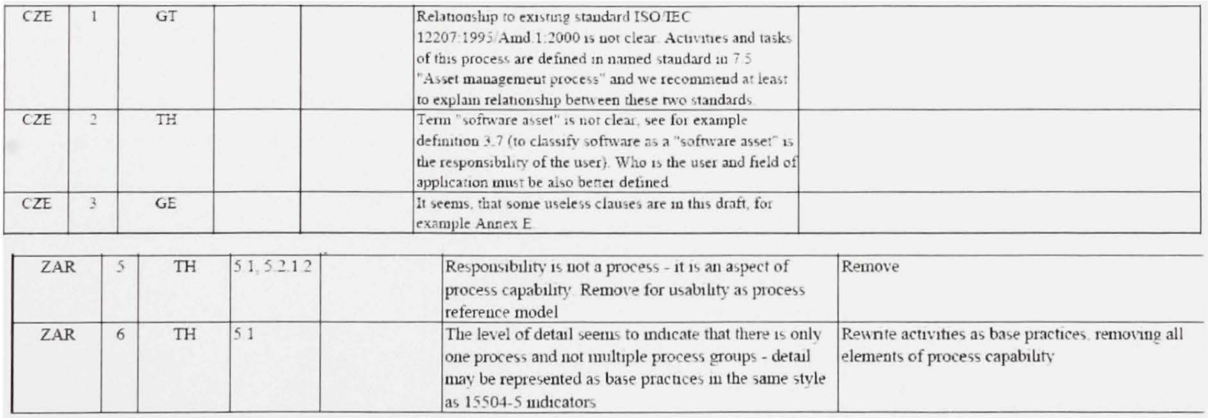
APPENDIX XVIII

LETTER BALLOT SUMMARY - N3169 - CD19770-1.3 - INFORMATION TECHNOLOGY – SOFTWARE ASSET MANAGEMENT PROCESSES (N3221)

TABLE OF VOTING AND COMMENTS RECEIVED



	"P" members	Approve	Disapprove	Abstain	Comments	Not voting
1.	Australia	X				
2.	Belgium					
3.	Brazil					
4.	Canada	X				
5.	China					
6.	Czech Republic		X		X	
7.	Denmark	X				
8.	Finland					
9.	France					
10.	Germany			X		
11.	Hungary					
12.	Ireland			X		
13.	Israel					
14.	Italy	X				
15.	Japan	X			X	
16.	Korea	X				
17.	Luxemburg					
18.	Netherlands					
19.	Portugal					
20.	Romania					
21.	South Africa		X		X	
22.	Spain	X				
23.	Sweden	X			X	
24.	Switzerland			X		
25.	Thailand					
26.	UK		X		X	
27.	Ukraine					
28.	USA			X		
29	**TOTAL**	**8**	**3**	**4**	**5**	**14**



CZE	1	GT			Relationship to existing standard ISO/IEC 12207:1995/Amd.1:2000 is not clear. Activities and tasks of this process are defined in named standard in 7.5 "Asset management process" and we recommend at least to explain relationship between these two standards.	
CZE	2	TH			Term "software asset" is not clear, see for example definition 3.7 (to classify software as a "software asset" is the responsibility of the user). Who is the user and field of application must be also better defined.	
CZE	3	GE			It seems, that some useless clauses are in this draft, for example Annex E.	

ZAR	5	TH	5.1, 5.2.1.2		Responsibility is not a process - it is an aspect of process capability. Remove for usability as process reference model	Remove
ZAR	6	TH	5.1		The level of detail seems to indicate that there is only one process and not multiple process groups - detail may be represented as base practices in the same style as 15504-5 indicators	Rewrite activities as base practices, removing all elements of process capability

APPENDIX XIX

MEETING MINUTES, WG21, SOFTWARE ASSET MANAGEMENT, HELSINKI, FINLAND, MAY 23-27 2005 (N3281)

3.2. Resolutions

Two resolutions:

JTC1/SC7 thus instructs its Secretariat to conduct an FCD ballot of the document WG21 N052 - ISO/IEC 19770 - 1: Software Asset Management Processes, when it is received by the secretariat with the appropriate documentation. (All documents are delivered.)

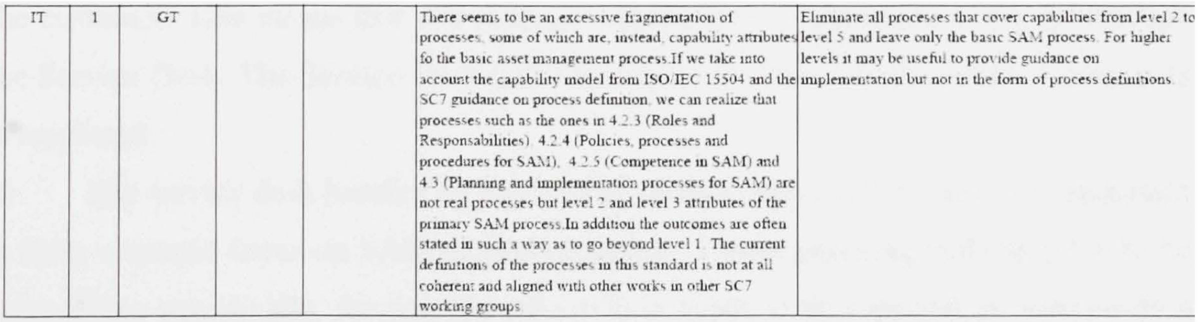
JTC1/SC7 instructs its Secretariat to issue a combined WD, CD Registration and CD ballot for the document WG21 Nxxx - ISO/IEC 19770-2: Software Asset Management Tag, when it is received by the Secretariat with the appropriate documentation. (Will be delivered after the interim meeting in Bari.)

APPENDIX XX

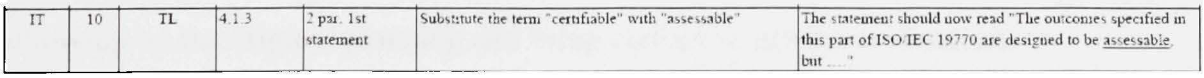
LETTER BALLOT SUMMARY, FCD 19770-1 - INFORMATION TECHNOLOGIES – SOFTWARE ASSET MANAGEMENT PROCESSES (N3375)

TABLE OF VOTING AND COMMENTS RECEIVED

	"P" members	Approve	Disapprove	Abstain	Comments	Not voting
1.	Australia	X				
2.	Belgium					X
3.	Brazil					X
4.	Canada	X				
5.	China					X
6.	Czech Republic	X				
7.	Denmark	X				
8.	Finland					X
9.	France	X			X	
10.	Germany	X				
11.	Iran					X
12.	Ireland			X		
13.	Israel					X
14.	Italy		X		X	
15.	Japan		X		X	
16.	Korea	X				
17.	Luxemburg					X
18.	Netherlands					X
19.	New Zealand					X
20.	Portugal					X
21.	Romania					X
22.	South Africa	X				
23.	Spain			X		
24.	Sweden	X			X	
25.	Switzerland			X		
26.	Thailand			X		
27.	UK	X			X	
28.	Ukraine	X				
29.	USA					X
29	**TOTAL**	**11**	**2**	**4**	**5**	**12**



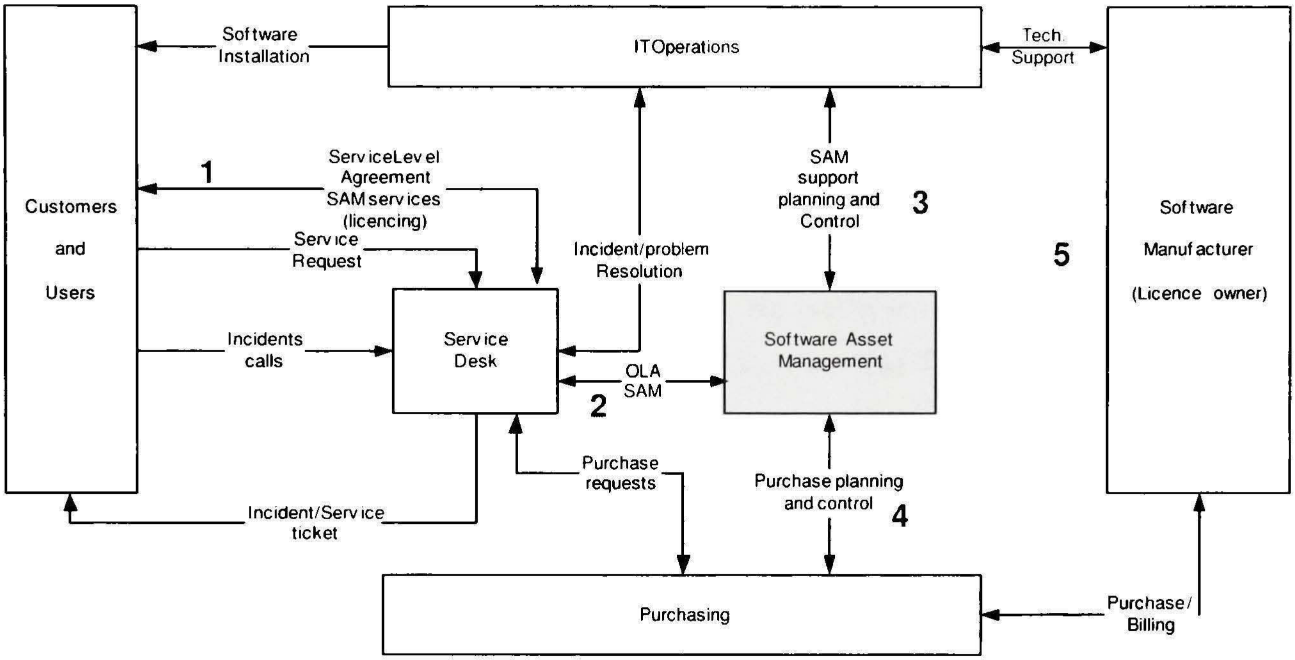
IT	1	GT			There seems to be an excessive fragmentation of processes, some of which are, instead, capability attributes fo the basic asset management process.If we take into account the capability model from ISO/IEC 15504 and the SC7 guidance on process definition, we can realize that processes such as the ones in 4.2.3 (Roles and Responsabilities), 4.2.4 (Policies, processes and procedures for SAM), 4.2.5 (Competence in SAM) and 4.3 (Planning and implementation processes for SAM) are not real processes but level 2 and level 3 attributes of the primary SAM process.In addition the outcomes are often stated in such a way as to go beyond level 1. The current definitions of the processes in this standard is not at all coherent and aligned with other works in other SC7 working groups.	Eliminate all processes that cover capabilities from level 2 to level 5 and leave only the basic SAM process. For higher levels it may be useful to provide guidance on implementation but not in the form of process definitions.



IT	10	TL	4.1.3	2 par. 1st statement	Substitute the term "certifiable" with "assessable"	The statement should now read "The outcomes specified in this part of ISO/IEC 19770 are designed to be assessable, but ...."

APPENDIX XXI

THE SOFTWARE ASSET MANAGER WITHIN THE IT ORGANIZATION



The five interfaces:

1) Following the ITIL philosophy, the service desk is the single point of contact for the customer. This means that any request, and thus any SAM request, will go through the Service Desk. The Service Desk will also ensure that any Service Level Agreements is respected.

2) The service desk handles all kinds of IT related requests, for SAM, it is important to have a special focus on SAM related processes if some planning and control is to be done. More specifically, for specific information needs to be captured in order to do a follow-up on the original planning and bring corrective actions as required.

3) Any Software Asset needs an IT environment with specific technical requirements such as the Hardware specifications, a specific Operating System and personnel to support the application. The number and the frequency of use of all these

resources needs to be planned in order to ensure that Service Level Agreements with customers are meet.

4) For each business needs, they are several software manufacturer available and most software manufacturers have more than one licence scheme, the choices made can have very important financial implications. This planning can be done by following directives from upper management, the data collected from the Service Desk in terms of software usage and demands as well as technical requirements and feedback from IT Operations.

5) The two main interactions with the Software manufacturer can be classified as financial or technical. The technical interaction comes from IT operations and is required to install and operate adequately each of the software on the customer's systems, which has sometime a very complex architecture. The financial interaction comes for purchasing/finance, which buys and purchase the software. To ensure that the correct version of the software is bought, coordination with the customer and IT operations is required. The choice of the software and its version may also be influenced by historical data on past costs, user appreciation and operational requirements.

APPENDIX XXII

MATURITY ASSESSMENT OF SAM MANAGEMENT/ORGANIZATION PROCESSES

Table 29

Organizational processes assessment results

#	Name	Purpose	Outcome	Incomplete	Performed	Managed	Defined	Quantitatively managed	Optimizing
4.2.2	**Corporate governance process for SAM**	The purpose of the *Corporate governance process for SAM* is to ensure that responsibility for management of software assets is recognized at the level of the corporate board or equivalent body, and that appropriate mechanisms are in place to ensure the proper discharge of this responsibility.	• There is a clear corporate statement for the purposes of this part of ISO/IEC 19770						
			• Responsibility for corporate governance of software and related assets is formally recognized by the corporate board or equivalent body.						
			• Corporate governance regulations or guidelines which are relevant to the organization for its use of software and related assets, in all countries where it operates, have been identified and documented, and are reviewed at least annually.						
			• An assessment of the risks associated with software and related assets, and management-specified mitigation approaches, is						

#	Name	Purpose	Outcome	Incomplete	Performed	Managed	Defined	Quantitatively managed	Optimizing
			documented, updated at least annually, and approved by the corporate board or equivalent body, covering at least the following						
4.2.3	**Roles and responsibilities for SAM**	The purpose of establishing *Roles and responsibilities for SAM* is to ensure that the roles and responsibilities for software and related assets are clearly defined, maintained and understood by all personnel potentially affected.	• The role of the SAM owner, responsible for corporate governance of software and related assets for the entire organization, is clearly defined and approved by the corporate board or equivalent body.						
			• Local roles and responsibilities for corporate governance of software and related assets are documented and assigned to specified individuals						
			• These responsibilities are communicated to all parts of the organization involved in any way with SAM, in the same way as other organization-wide and local policies are communicated.						
4.2.4	**Policies, processes and procedures for SAM**	The purpose of *Policies, processes and procedures for SAM* is to ensure that an organization maintains clear	• There is a structured approach to creating, reviewing, approving, issuing, and controlling policies, processes, procedures and related documentation relevant to SAM so that it is always possible to determine the complete set available, which version of each document is currently in effect and which documents apply to different types of software and related						
		policies, processes and procedures to ensure effective planning, operation and control of							

#	Name	Purpose	Outcome	Incomplete	Performed	Managed	Defined	Quantitatively managed	Optimizing
		SAM.	assets						
			• Policy, process and procedure documentation required by this part of ISO/IEC 19770 is organized by the process classifications of this part of ISO/IEC 19770 or with a cross-reference to these classifications.						
			• Policies are developed, approved and issued covering at a minimum:						
			○ 1) Individual and corporate responsibilities for corporate governance of software and related assets.						
			○ 2) Any restrictions on personal use of corporate software and related assets.						
			○ 3) Requirement for compliance with legal and regulatory requirements, including for copyright and data protection.						
			○ 4) Any requirement for approvals for installation or use of software, whether purchased or not.						
			○ 5) Disciplinary implications of violation of these policies.						
			• Policies and procedures are communicated to all employees in a way which (a) reaches all new employees when they start, and continuing employees at least annually; (b) requires positive acknowledgement back from employees when they start and at least annually; and (c) is readily accessible at all times to employees.						
4.2.5	**Competence in SAM**	The purpose of the *Competence in SAM* process is to ensure that appropriate	• A review is documented and updated at least annually which covers the availability and uptake of training and certification by						

#	Name	Purpose	Outcome	Incomplete	Performed	Managed	Defined	Quantitatively managed	Optimizing
		competence and expertise in SAM is available and is being applied.	personnel with SAM responsibilities for:						
			○ SAM in general.						
			○ Licensing for software manufacturers whose software is being used.						
			• A review is documented and updated at least annually which determines the availability of licensing guidance checklists and training made available by software manufacturers whose software is being used						
			• Licensing guidance checklists, made available by software manufacturers whose software is being used, are completed and updated at least annually, and signed off by appropriate management						
			• Personnel with SAM management responsibilities receive training in SAM and in relevant licensing, including both initial training and formal continuing education annually						
4.3.2	**Planning for SAM**	The purpose of *Planning for SAM* is to ensure appropriate preparation and planning for the effective and efficient accomplishment of SAM objectives	• Management objectives for SAM are developed and proposed for approval by the corporate board or equivalent body, and updated at least annually						
			• A plan (the 'SAM plan') for implementing and delivering SAM is developed						

#	Name	Purpose	Outcome	Incomplete	Performed	Managed	Defined	Quantitatively managed	Optimizing
			and documented, and updated						
			• at least annually, which includes:						
			○ 1) A clear scope statement ('software asset scope') describing which types of software are included; the coverage of related assets, including any beyond the minimum required by this part of ISO/IEC 19770; and any interfaces with or requirements for other organizations or systems.						
			○ 2) A clear specification of which policies, processes and procedures are required for assets in scope.						
			○ 3) A clear explanation of the approach to managing, auditing and improving SAM including automation as appropriate to support the processes.						
			○ 4) An explanation of the approach to be used to identifying, assessing and managing issues and risks related to the achievement of the defined management objectives.						
			○ 5) Schedules and responsibilities for periodic activities, including preparation of management reports and performance of verification and compliance activities.						
			○ 6) Identification of the resources including budget needed to implement the SAM plan.						
			○ 7) Measurable targets for tracking progress against the SAM plan, including target measures for accuracy of the asset management records.						

#	Name	Purpose	Outcome	Incomplete	Performed	Managed	Defined	Quantitatively managed	Optimizing
4.3.3	**Implementation of SAM**	The purpose of *Implementation of SAM* is to accomplish overall SAM objectives and the SAM plan.	• Mechanisms are in place to collect information, including from local SAM owners, about changes, issues and risks that affect the SAM plan throughout the year.						
			• Regular status reports (at least quarterly) are prepared by the SAM owner detailing the overall progress against the SAM plan for reporting to the corporate board or equivalent body.						
			• Follow-up on any variances identified takes place promptly and is documented.						
4.3.4	**Monitoring and review of SAM**	The purpose of *Monitoring and review of SAM* is to ensure that the management objectives for SAM are being achieved.	• A formal review is conducted at least annually:						
			○ to assess whether management objectives for SAM and the SAM plan are being achieved						
			○ to summarize performance against all performance measures specified in the SAM plan and in service level agreements related to SAM						
			○ to provide a summary of the findings of the *Conformance verification for SAM* process						
			○ to conclude on the basis of the above whether:						
			▪ the policies approved by management which are relevant for SAM have been effectively disseminated throughout the organizational						

#	Name	Purpose	Outcome	Incomplete	Performed	Managed	Defined	Quantitatively managed	Optimizing
			scope defined for the purposes of this part of ISO/IEC 19770						
			▪ the processes and procedures which are relevant for SAM, as approved by management, have been effectively implemented throughout the organizational scope defined for the purposes of this part of ISO/IEC 19770						
			○ to summarize any exceptions identified and actions which may need to be taken as a result of the above						
			○ to identify opportunities for improvement in the provision of services for software and related assets						
			○ to consider whether there is a need for a review of policies, processes and procedures as to their continued appropriateness, completeness and correctness.						
			• b) The SAM owner signs off on the report, documents decisions and actions that are to be taken as a result, and copies it to the corporate board or equivalent body						
			• c) There is a periodic review (at least annually) of whether software and related assets are deployed in the most cost-effective manner possible; and recommendations are made for possible improvement.						
4.3.5	**Continual improvement of SAM**	The purpose of *Continual improvement of SAM* is to ensure that opportunities for improvement are identified	• A mechanism is in place to collect and record suggested improvements in SAM arising from all sources throughout the year.						

#	Name	Purpose	Outcome	Incomplete	Performed	Managed	Defined	Quantitatively managed	Optimizing
		and acted upon where considered justified, both in the use of software and related assets and in the SAM processes themselves.							
			• Suggestions for improvement are periodically assessed, prioritized and approved for incorporation in SAM implementation and improvement plans.						

APPENDIX XXIII

MATURITY ASSESSMENT OF SAM OPERATIONAL PROCESSES

Table 30

Operational processes assessment results

#	Name	Purpose	1. Outcome	Incomplete	Performed	Managed	Defined	Quantitatively managed	Optimizing
4.4.2	**Software asset identification**	The purpose of *Software asset identification* is to ensure that the necessary classes of assets are selected and grouped; and defined by appropriate characteristics that enable effective and efficient control of software and related assets.	28. Items to be managed are chosen using established selection criteria, grouped, classified and identified to ensure that they are manageable and traceable throughout their lifecycle.						
			29. Items to be managed include: a. All platforms on which software can be installed or run b. Software definitive master versions and distribution copies c. Software builds and releases						

#	Name	Purpose	1. Outcome	Incomplete	Performed	Managed	Defined	Quantitatively managed	Optimizing
			(originals and distribution copies) d. All installed software e. Software versions f. Patches and updates g. Licenses including underlying licenses and effective full licenses h. Proof of license documentation i. Contracts (including terms and conditions) relating to software assets, including both hard-copy and electronic j. Both physical and electronic stores of the above, as relevant k. Licensing models						
			30. Software should be manageable both by files and by packages corresponding to specific products released by software manufacturers or developers						
			31. Basic information required for all assets is l. Unique identifier m. Name/description n. Location o. Custodianship (or owner) p. Status q. Version (where applicable)						
			32. A register of stores and inventories exists, clarifying which stores and types of information are held, with duplication allowed only if duplicate information can be traced back to the definitive source record						
4.4.3	**Software asset**	The purpose of *Software*	33. Policies and procedures are developed, approved and issued which include the						

#	Name	Purpose	1. Outcome	Incomplete	Performed	Managed	Defined	Quantitatively managed	Optimizing
	inventory management	*asset inventory management* is to ensure that physical instances of software assets are properly stored; and that required data about characteristics for all assets and configuration items is accurately recorded throughout the life cycle. It also provides information on software assets and related assets to support the effectiveness and efficiency of other business processes.	management and maintenance of inventories and physical/electronic stores including access controls which: 1) protect them from unauthorized access, 2) change or corruption. provide a means for disaster recovery. 3) underlying licenses and effective full licenses held						
			34. Inventories exist of 1) all platforms on which software assets can be installed or run. 2) all authorized						

#	Name	Purpose	1. Outcome	Incomplete	Performed	Managed	Defined	Quantitatively managed	Optimizing
			installed software showing (a) packages and versions which can be individually licensed or authorized for deployment; and (b) update/patch status of software; all by platform on which installed.						
			35. Inventories and corresponding physical/electronic stores exist of 1) software (definitive master versions and distribution copies) 2) software builds and releases (originals and distribution copies) 3) contracts relating to software assets, both hard-copy and electronic 4) proof of license documentation.						
			36. Inventories or other clearly defined analysis or metric mechanisms exist to determine any licensing usage based on criteria other than software installations.						
			37. e) Arrangements are made to ensure the continued availability of the sources listed above.						
			38. f) Each inventory report produced has a clear description including its identity, purpose, and details of the data source.						
4.4.4	**Software asset control**	The purpose of *Software asset control* is to provide the control mechanism over software assets and	39. a) An audit trail is maintained of changes made to software and related assets including changes in the status, location, custodianship and version						

#	Name	Purpose	1. Outcome	Incomplete	Performed	Managed	Defined	Quantitatively managed	Optimizing
		changes to software and related assets while maintaining a record of changes to status and approvals.[1]							
			40. b) Policies and procedures are developed, approved and issued for the development, maintenance and management of software versions, images/builds and releases.						
			41. c) Policies and procedures are developed, approved and issued which require that a baseline of the appropriate assets is taken before a release of software to the live environment in a manner that can be used for subsequent checking against actual deployment.						
4.5.2	**Software asset record verification**	The purpose of *Software asset record verification* is to ensure that records reflect accurately and completely what they are supposed to record, and conversely that what they	42. Procedures are developed, approved and issued for the *Software asset record verification* process to include: 1) At least quarterly there is a reconciliation between what is installed on each platform and what was authorized for installation, including reporting on exceptions identified. 2) The hardware inventory including locations is verified at least 6-monthly, including reporting on exceptions identified.						

#	Name	Purpose	1. Outcome	Incomplete	Performed	Managed	Defined	Quantitatively managed	Optimizing
		record has not changed without approval.	3) The inventory of software programs (definitive master versions and distribution copies) is verified at least 6-monthly, including reporting on exceptions identified. 4) The inventory of software builds (originals and distribution copies) is verified at least 6-monthly, including reporting on exceptions identified. 5) The physical store of proof of license documentation is verified (including for authenticity) at least annually, including reporting on exceptions identified. 6) The bases for and calculations of effective licenses from underlying licenses are reviewed at least annually, to ensure that necessary underlying licenses exist and that quantities are not being double counted. 7) The physical store of contractual documentation related to software assets is verified for completeness at least annually, including reporting on exceptions identified. 8) The contracts inventory is verified at least annually, including reporting on exceptions identified. 9) Follow-up corrective actions on any discrepancies identified above take place and are documented.						
4.5.3	**Software**	The purpose	43. Procedures are developed, approved and issued for the						

#	Name	Purpose	1. Outcome	Incomplete	Performed	Managed	Defined	Quantitatively managed	Optimizing
	licensing compliance	of the *Software licensing compliance* process is to ensure that all intellectual property used by the organization but owned by others, pertaining to software and related assets, is properly licensed and used in accordance with its terms and conditions.	*Software licensing compliance* process to include the following: Reconciliation is conducted at least quarterly between effective licenses owned and licenses required for software used, taking into account the way licensing requirements are determined per license terms and conditions. 2) Discrepancies identified in this reconciliation are promptly recorded, analyzed and the root cause is determined. 3) Follow up actions are prioritized and executed.						
4.5.4	**Software asset security compliance**	The purpose of *Software asset security compliance* is to ensure that security requirements related to the use of software and related assets are complied with.	44. Actual practice against policy is reviewed at least annually.						
			45. Follow-up on any discrepancies identified in this review takes place and is documented						

#	Name	Purpose	1. Outcome	Incomplete	Performed	Managed	Defined	Quantitatively managed	Optimizing
4.5.5	**Conformance verification for SAM**	The purpose of *Conformance verification for SAM* is to ensure that there is continuing compliance with the requirements of this part of ISO/IEC 19770 including compliance with required policies and procedures.	46. Policies and procedures are developed, approved and issued for verifying compliance with this part of ISO/IEC 19770, which ensure verification at least on a sample basis annually against all of the requirements specified in this part of ISO/IEC 19770. This shall include verification that that procedures implemented by the organization for other SAM processes are meeting all requirements specified in this part of ISO/IEC 19770 for those procedures.						
			47. Documentary evidence exists that demonstrates (a) that the verification procedures above are being performed, and (b) that corrective follow-up action is taken until successful completion on the causes of all identified exceptions						
4.6.2	**Relationship and contract management for SAM**	The purpose of *Relationship and contract management for SAM* is to manage relationships with other organizations, both external and internal,	48. Policies and procedures are developed, approved and issued for managing relationships with suppliers providing software and related assets and services, to include 1) Definitions of responsibilities for supplier management with individuals assigned to have clear overall responsibility for managing each supplier. 2) Developing invitations to tender for the supply of software or related services; to						

#	Name	Purpose	1. Outcome	Incomplete	Performed	Managed	Defined	Quantitatively managed	Optimizing
		to ensure the provision of seamless, quality SAM services, and to manage all contracts for software and related assets and services.	ensure that the process includes consideration of requirements for SAM, including service level management, security controls, release and change management. r. 3) Formal documented reviews at least 6-monthly of supplier performance, achievements and issues, with documented conclusions and decisions about any actions to be taken.						
			49. Policies and procedures are developed, approved and issued for managing customer-side relationships, to include: 1) Definitions of responsibilities for managing customer-side business relationships with respect to software and related assets and services. 2) A formal review at least annually of current and future software requirements of customers and the business as a whole. 3) Formal documented reviews at least annually of service provider performance, customer satisfaction, achievements and issues, with documented conclusions and decisions about any actions to be taken.						
			50. Policies and procedures are developed, approved and issued for managing contracts, to include: Ensuring that						

#	Name	Purpose	1. Outcome	Incomplete	Performed	Managed	Defined	Quantitatively managed	Optimizing
			contractual details are recorded in an on-going contract management system as contracts are signed. Holding copies of all signed contractual documentation securely with copies kept in a document management system. Documented reviews at least 6-monthly and also prior to contract expiry, of all contracts for software and related assets and services, with documented conclusions and decisions about any actions to be taken.						
4.6.3	**Financial management for SAM**	The purpose of *Financial management for SAM* is budgeting and accounting for software and related assets; and ensuring that relevant financial information is readily available for financial reporting, tax planning, and calculations such as total cost of ownership and return on investment.	51. Definitions of financial information relevant to the management of software and related assets are agreed with relevant parties and documented by asset type.						

#	Name	Purpose	1.	Outcome	Incomplete	Performed	Managed	Defined	Quantitatively managed	Optimizing
			52.	b) Formal budgets are developed for the acquisition of software assets (externally or internally) and the related support and infrastructure costs						
			53.	c) Actual expenditure on software assets and the related support and infrastructure costs is accounted for against budget.						
			54.	d) Clearly documented financial information is readily available about software asset values (including historical cost and depreciated cost).						
			55.	e) There are formal documented reviews at least quarterly of actual expenditure against budget, with documented conclusions and decisions about any actions to be taken.						
4.6.4	**Service level management for SAM**	The purpose of *Service level management for SAM* is to define, record and manage levels of service related to SAM.	56.	Service level agreements and supporting agreements are developed and approved for services that are performed within the scope of SAM; to include that: 1) Services relating to software acquisition, installation, moves, and changes of software assets and related assets are defined and agreed with relevant parties together with the corresponding service level targets and workload characteristics. 2) The customer and user obligations and responsibilities in relation to SAM are defined or referenced from the service level agreement.						
			57.	Actual workloads and service levels against targets for SAM are reported regularly (at least quarterly), and the reasons for non-						

#	Name	Purpose	1. Outcome	Incomplete	Performed	Managed	Defined	Quantitatively managed	Optimizing
			conformance are documented.						
			58. Regular reviews (at least quarterly) by the relevant parties are held to review the service levels for SAM with documented conclusions and decisions about any actions to be taken.						
4.6.5	**Security management for SAM**	The purpose of *Security management for SAM* is to manage information security effectively within all SAM activities and support the approval requirements related to SAM	59. a) A formal policy is developed and approved regarding security/access restrictions to all SAM resources, including physical/electronic stores of software, software builds and releases.						
			60. b) Access controls are specified, both physical and logical, to enforce the approval requirements of SAM policies						
			61. c) There is documentary evidence that these specified access controls are being implemented in practice.						
4.7.2	**Change management process**[2]	The purpose of the *Change management*	62. 1) All change requests that affect software or related assets or services, or SAM processes, are identified and recorded.						

[2] The Change management process with respect to software and related assets is tightly linked to the Software
asset control process, which provides the control mechanism underlying any changes to be made to software and related assets.

#	Name	Purpose	1. Outcome	Incomplete	Performed	Managed	Defined	Quantitatively managed	Optimizing
		process with respect to software and related assets is to ensure that all changes which impact on SAM are assessed, approved, implemented and reviewed in a controlled manner and meet all record-keeping requirements.							
			63. 2) Change requests affecting software or related assets or services, or SAM processes, are assessed for possible impacts, prioritized, and approved by the responsible management.						
			64. 3) The process implementing the approved change request does so only in accordance with the approval.						
			65. 4) All changes affecting software or related assets or services, or SAM processes, are recorded.						
			66. 5) The success or failure of such changes is documented and periodically reviewed.						
4.7.3	**Acquisition process**	The purpose of the *Acquisition process* in respect of software and	67. a) Standard architectures are defined for the provision of software services, as are the criteria for deviating from those standards.						

#	Name	Purpose	1. Outcome	Incomplete	Performed	Managed	Defined	Quantitatively managed	Optimizing
		related assets is to ensure that they are acquired in a controlled manner and properly recorded.							
			68. b) Standard software configurations are defined, as are the criteria for deviating from those standards.						
			69. c) Policies and procedures are developed, properly authorized and issued for requisitioning and ordering software assets and related assets, including: 1) How requirements are specified. 2) Management and technical approvals required. 3) Use/redeployment of existing licenses if available. 4) Recording future purchase requirements in those cases where software can be deployed before reporting and payment.						
			70. Policies and procedures are developed, properly authorized and issued for receipt-processing functions related to software and related assets, including 1) Processing invoices, including reconciliations to orders and retention of copies for license management purposes. 2) Ensuring the receipt and safe-keeping of valid proof of license for all licenses purchased.						
			71. Processing incoming media which includes requirements						

#	Name	Purpose	1.	Outcome	Incomplete	Performed	Managed	Defined	Quantitatively managed	Optimizing
				for verification, record-keeping and safekeeping of contents (physical media and electronic copies).						
4.7.4	**Software development process**	The purpose of the *Software development process* in respect of software and related assets is to ensure that they are developed in a way which considers SAM requirements.	72.	There is a formal process for software development ensuring the following have been considered 1) Standard architectures and standard configurations. 2) Licence constraints and dependencies.						
			73.	There is a formal process for software development ensuring that: 1) Software products are placed under software asset control. 2) A plan is developed for how software is to be released, and deployed						
4.7.5	**Software release management process**	The purpose of the *Software release management process* in respect of software and related assets is to ensure that releases are planned and executed	74.	1) A controlled acceptance environment is used to build and test all proposed releases including patches prior to release.						

#	Name	Purpose	1. Outcome	Incomplete	Performed	Managed	Defined	Quantitatively managed	Optimizing
		in a way which supports SAM requirements.							
			75. 2) The frequency and type of releases are planned and agreed with the business and customers, including the frequency of security patch releases.						
			76. 3) The planned release dates and deliverables are recorded with references to related change requests and problems, and communicated to incident management. 77.						
			78. 4) The release of software and related assets is approved by the responsible management						
			79. 5) The success or failure of releases is recorded, and periodically reviewed.						
4.7.6	**Software deployment process**	The purpose of the *Software deployment process* in respect of SAM is to ensure that software deployment and redeployment is executed in a way which supports SAM requirements.	80. 1) The distribution of software and related assets is approved by the responsible management.						
			81. 2) For any deployment there is a back out procedure or method of remediation if the deployment is not successful						
			82. 3) Security requirements are						

#	Name	Purpose	1.	Outcome	Incomplete	Performed	Managed	Defined	Quantitatively managed	Optimizing
				complied with, including over access to the software being distributed and after it is installed						
			83.	4) All changes to status of the relevant software and related assets are recorded accurately and on a timely basis, including any change of custodianship for the assets.						
			84.	5) There is a documented control to verify that what was deployed is the same as what was authorized to be deployed.						
			85.	6) The success or failure of deployments is recorded, and periodically reviewed.						
4.7.7	**Incident management process**	The purpose of the *Incident management process* in respect of software and related assets is to monitor and respond to incidents in ongoing operations relevant to software and related assets.	86.	1) All incidents that affect software or related assets or SAM processes are recorded and classified as to their priority for resolution.						
			87.	2) All such Incidents are resolved in accordance with their priority for resolution, and the resolution is documented.						
4.7.8	**Problem management process**	The purpose of the *Problem management process* in	88.	1) All incidents that affect software or related assets or services or SAM processes are recorded and classified as to their impact.						

#	Name	Purpose	1. Outcome	Incomplete	Performed	Managed	Defined	Quantitatively managed	Optimizing
		respect of software and related assets is to keep software assets current and in operational fitness, including through proactive identification and analysis of the cause of incidents and addressing the underlying problems.							
			89. 2) Higher priority incidents are analyzed for the underlying causes and prioritized for resolution.						
			90. 3) Underlying causes are documented and communicated to incident management.						
			91. 4) Problems are resolved in accordance with their priority for resolution, and the resolution is documented and communicated to incident management.						
4.7.9	**Retirement process**	The purpose of the *Retirement process* in respect of software and related assets is to remove	92. 1) Deployed copies of software are removed from retired hardware.						

#	Name	Purpose	1. Outcome	Incomplete	Performed	Managed	Defined	Quantitatively managed	Optimizing
		software and related assets from use, including recycling of associated assets where appropriate, in accordance with company policy and meeting all record-keeping requirements.							
			93. 2) Licenses which can be redeployed are identified for redeployment.						

REFERENCES

[1] P. Adams, "IT Asset Auto-Discovery and Inventory Tool Q&A," Gartner Select Q&A, QA-11-7354, 16 August 2000 2000.

[2] P. Adams, "Understanding the Key Attributes of IT Asset Management," Gartner Markets, M-17-0473, 16 July 2002 2002.

[3] P. Adams, "Workflow Processes for Tracking and Buying Your IT Assets," Gartner Tutorials, TU-15-0511, 10 April 2001 2001.

[4] P. Adams and R. Colville, "IT Assets, Inventory and Configuration: How Do They Differ?," GartnerGroup RAS Services 12 March 2001 2001.

[5] P. Adams and R. Colville, "ITIL Process Gaps for Asset and Configuration Data," Gartner COM-16-7539, 6 June 2002 2002.

[6] K. Brittain and P. Adams, "Asset Management in a CSD: Square Peg in a Round Hole?," Gartner Decision Framework, DF-12-9543, 21 June 2001 2001.

[7] M. Margevicius and L. Mieritz, "Gaining Efficiencies With PC Life Cycle Management," Gartner Tactical Guidelines, TG-14-5059, 30 October 2001 2001.

[8] F. O'Brien, "PC Disposal: Data Security and Sanitization Responsibilities," Gartner Tactical Guidelines, TG-15-6943, 2 April 2002 2002.

[9] F. O'Brien, "PC Disposal: Methods for Secure Data Sanitization," Gartner Tutorials, TU-15-8513, 2 April 2002 2002.

[10] M. Truglio, "IT Asset Management: Closed Loop or Flat Line?," Gartner Strategic Planning, SPA-09-6332, 11 January 2000 2000.

[11] D. Vogel, "Configuration Management: An IT Operational Foundation," META Group September 2002 2002.

[12] R. McNeill, "Common Mistakes in IT Asset Management," Giga Information Group June 27, 2002 2002.

[13] R. McNeill and J. Ragsdale, "The Total Economic Impact™ of Integrated IT Asset Management and Help Desk," Giga Information Group August 6, 2002 2002.

[14] Central Computer & Telecommunications Agency, *IT Infrastructure Library - Service Support*: Office of Government Commerce, 2000.

[15] Central Computer & Telecommunications Agency, *IT Infrastructure Library - Service Delivery*: Office of Government Commerce, 2000.

[16] Business Software Alliance BSA, "Asset Management Resources," in *http://www.bsa.org/usa/antipiracy/Asset-Management-Resources.cfm*, vol. 2005, 2005.

[17] Microsoft Corporation, "Microsoft Software Asset Management," in *http://www.microsoft.com/resources/sam/default.mspx*, vol. 2006, 2006.

[18] J. Fontana, "Manage your assets, Network World," in *http://www.networkworld.com/careers/2002/0805man.html*, 2002.

[19] Office of Government Commerce, *Software Asset Management (It Infrastructure Library Services)*: Stationery Office, 2003.

[20] J.-N. Lee, M. Q. Huynh, R. C.-W. Kwok, and S.-M. Pi, "IT outsourcing evolution- Past, present, and future," *Communications of the ACM*, vol. 46, pp. 84-89, 2003.

[21] P. Bourque, F. Robert, J. M. Lavoie, A. Lee, S. Trudel, and T. C. Lethbridge, "Guide to the Software Engineering Body of Knowledge (SWEBOK) and the Software Engineering Education Knowledge (SEEK) - a preliminary mapping," presented at Proceedings 10th International Workshop on Software Technology and Engineering Practice, 6-8 Oct. 2002, Montreal, Que., Canada, 2003.

[22] S. L. Pfleeger, *Software engineering theory and practice*, 2nd ed. Upper Saddle River, N.J.: Prentice-Hall, 2001.

[23] R. S. Pressman, *Software engineering a practitioner's approach*, 5th ed. New York, N.Y.: McGraw-Hill, 2001.

[24] D. J. Reifer, *Software management*, 6th ed. ed. Washington, D.C.: IEEE Computer Society Press, 2002.

[25] I. Sommerville, *Software engineering*, 6th ed. Reading, Mass.: Addison-Wesley, 2001.

[26] S. Heilbronner and R. Wies, "Managing PC networks," *IEEE Communications Magazine*, vol. 35, pp. 112-17, 1997.

[27] C. Yang and J.-B. Huang, "A decision model for IS outsourcing," *International Journal of Information Management*, vol. 20, pp. 225-39, 2000.

[28] M. Turner and A. Smith, "It outsourcing: The challenge of changing technology in it outsourcing agreements," *Computer Law and Security Report*, vol. 18, pp. 181-186, 2002.

[29] M. S. Krishnan, "Software release management: a business perspective," presented at Proceedings of the 1994 conference of the Centre for Advanced Studies on Collaborative research, 1994.

[30] S. H. Kwok, "Digital rights management for the online music business," *ACM SIGecom Exchanges*, vol. Volume 3 Issue 3, 2002.

[31] R. C. Hauser, "Does licensing require new access control techniques?." Fairfax, Virginia, United States: ACM Press, 1993, pp. 1--8.

[32] L. Ling-feng, H. Pei-qing, Y. Qing-ding, and Z. Xiao-min, "Research of fuel inventory management in powerplant," *Industrial Engineering and Management*, vol. 8, pp. 67-71, 2003.

[33] Software-Engineering-Institute, "Capability Maturity Model® Integration (CMMI) ", vol. CMMI-SE/SW/IPPD/SS Version 1.1. Carnegie Mellon: CMMI-SE/SW/IPPD/SS Version 1.1, 2002.

[34] E. Cronin, S. Jamin, T. Malkin, and P. McDaniel, "On the performance, feasibility, and use of forward-secure signatures," in *Proceedings of the 10th ACM conference on Computer and communication security*. Washington D.C., USA: ACM Press, 2003, pp. 131--144.

[35] J. N. a. C. T. a. C. Collberg, "A functional taxonomy for software watermarking," in *Proceedings of the twenty-fifth Australasian conference on Computer science*. Melbourne, Victoria, Australia: Australian Computer Society, Inc., 2002, pp. 177--186.

[36] W. W. Royce, "Managing the Development of Large Software Systems," presented at Proceedings of IEEE WESCON, 1970.

[37] B. W. Boehm and K. J. Sullivan, "Software economics: a roadmap," presented at International Conference on Software Engineering, Limerick, Ireland, 2000.

[38] H. Reichgelt and V. Jovanovic, "Software Management as an Information Technology Knowledge Area," presented at CITC4'03, Lafayette, Indiana, USA., 2003.

[39] B. Barafort, B. D. Renzo, and O. Merlan, "Benefits resulting from the combined use of ISO/IEC 15504 with the Information Technology Infrastructure Library (ITIL)," presented at Product Focused Software Process Improvement. 4th International Conference, PROFES 2002. Proceedings, 9-11 Dec. 2002, Rovaniemi, Finland, 2002.

[40] International Organization for Standardization, "Software Engineering -- Software Life Cycle Processes -- Maintenance ", vol. ISO/IEC 14764:2006 Geneva, Switzerland: ISO/IEC 14764:2006 1998.

[41] Institute of Electrical and Electronics Engineers, "IEEE Recommended Practice for Software Acquisition," vol. IEEE-Std-1062: IEEE-Std-1062, 1998.

[42] R. J. Tersine and J. H. Campbell, *Principles of inventory and materials management*, 3rd ed. New York, N. Y.: Elsevier Science, 1988.

[43] P. E. Hicks, *Industrial engineering and management : a new perspective*, 2nd ed. ed. New York, N.Y.: McGraw-Hill, 1994.

[44] M. M. Naim, "The book that changed the world [inventory control]," *Manufacturing Engineer*, vol. 76, pp. 13-16, 1997.

[45] T. Y. Choi and J. K. Liker, "Guest Editorial - Supply Chain Management as an Emerging Focus of Technology Management," *IEEE Transactions on Engineering Management*, vol. 49, pp. 198-204, 2002.

[46] J. Rimell, "The importance of asset data to support an effective asset management policy," presented at IEE Seminar Integrating Technologies into Distribution Systems (Ref. No.1999/078), 13 May 1999, Birmingham, UK, 1999.

[47] J. C. Steed, "Aspects of how asset management can be influenced by modern condition monitoring and information management systems," presented at IEE Colloquium on HV Measurement, Condition Monitoring and Associated Database Handling Strategies (Ref. No.1998/448), 3 June 1998, London, UK, 1998.

[48] G. J. Anders, J. Endrenyi, and C. Yung, "Risk-based planner for asset management [of electric utilities]," *IEEE Computer Applications in Power*, vol. 14, pp. 20-6, 2001.

[49] S. Tohnai, T. Hirai, H. Ohkawa, and K. Matsuda, "Asset management system based on profile of field devices," presented at SICE 2002. Proceedings of the 41st SICE Annual Conference, 5-7 Aug. 2002, Osaka, Japan, 2002.

[50] B. Quak, E. Gulski, P. Wester, and J. J. Smit, "Fundamental aspects of information processing and the decision process to support asset management," presented at Proceedings of the 7th International Conference on Properties and Applications of Dielectric Materials, Jun 1-5 2003, Nagoya, Japan, 2003.

[51] J. C. Piepenbrink, "Clinical engineering at the new Boston City Hospital," presented at Proceedings of ELECTRO '94, 10-12 May 1994, Boston, MA, USA, 1994.

[52] V. Berko-Boateng and H. Hilbert, "Design for environment-an enabler for total asset management," presented at IEEE International Symposium on Electronics and the Environment, May 10-12 1993, Arlington, VA, USA, 1993.

[53] V. Berko-Boateng, J. Azar, E. De Jong, and G. A. Yander, "Asset recycle management. A total approach to product design for the environment," presented at IEEE International Symposium on Electronics and the Environment, May 10-12 1993, Arlington, VA, USA, 1993.

[54] M. J. Rogers, A. Gupta, and C. D. Maranas, "Real options based analysis of optimal pharmaceutical research and development portfolios," *Industrial and Engineering Chemistry Research*, vol. 41, pp. 6607-6620, 2002.

[55] Association of Local Government Engineering New Zealand and Public Works Engineering of Australia, "International Infrastructure Management Manual (IIMM)," vol. Version 2.0 ISBN 0-473-09137-2. Australia/New Zealand: Version 2.0 ISBN 0-473-09137-2, 2002.

[56] D. Déry, Abran, A, "Assets Management - Modeling Issues and Proposed Models " *Metrics News* vol. vol. 9 pp. pp. 39-46 2004

[57] J.-P. Jacquet and A. Abran, Walnut Creek, CA, USA.

[58] International Organization for Standardization, "Information technology -- Process assessment -- Part 1: Concepts and vocabulary," vol. ISO/IEC 15504-1:2004. Geneva, Switzerland: ISO/IEC 15504-1:2004, 2004, pp. 35.

[59] Software-Engineering-Institute, "CMMI Maturity Profile September 2006 Report ". Carnegie Mellon: http://www.sei.cmu.edu/appraisal-program/profile/pdf/CMMI/2006sepCMMI.pdf, 2006.

[60] International Organization for Standardization, "Information technology -- Process assessment -- Part 2: Performing an assessment ", vol. ISO/IEC 15504-2:2003. Geneva, Switzerland: ISO/IEC 15504-2:2003, 2003, pp. 16.

[61] International Organization for Standardization, "Information technology -- Process assessment -- Part 5:An exemplar Process Assessment Model determination," vol. ISO/IEC 15504-5:2005. Geneva, Switzerland: ISO/IEC 15504-5:2005, 2005, pp. 172.

[62] A. Stuart, "Meet SAM," in *RedMond Channel Partner Online*, 2006.

[63] D. Dery, "SAM Standards: A Review of ISO 19770-1 and ISO 19770-2," in *http://www.gelog.etsmtl.ca/publications/pdf-presentations/1000.pdf* Chicago, Illinois: ECPweb.com, 2006.

[64] D. Dery, "Future developments – Universal software asset tag," in *http://www.gelog.etsmtl.ca/publications/pdf-presentations/1020.pdf*, B. C. Series, Ed. BSI Group Headquarters, Chiswick, London, England, 2006.

[65] International Organization for Standardization, "JTC1/SC7 Published Standards," in

http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_tc_browse.htm?commid=45086&published=true, vol. 2007, March 2007.

[66] Institute of Electrical and Electronics Engineers, "IEEE Published Standards," in *http://ieeexplore.ieee.org/xpl/standards.jsp*, vol. 2007, March 2007.

[67] International Organization for Standardization, "Standard for Information Technology— Software life cycle processes," vol. ISO/IEC-12207. Geneva, Switzerland: ISO/IEC-12207, 1995.