

## An Itsm-Based Implementation Method to Maintain Software Assets in Order to Sustain Organizational Activities

Mohammad Sharifi<sup>1</sup>, Masarat Ayat<sup>2</sup>, Wan Khairuzzaman Wan Ismail<sup>3</sup>, Shamsul Sahibuddin<sup>4</sup>, Suhaimi Ibrahim<sup>4</sup>

<sup>1</sup>Faculty of Management and Human Resource Development, Universiti Teknologi Malaysia, Skudai, Johor, Malaysia

<sup>2</sup>Science and Research Branch of Islamic Azad University (SRBIAU), Tehran, Iran

<sup>3</sup>International Business School (IBS), Universiti Teknologi Malaysia, KL Campus, Malaysia

<sup>4</sup>Advanced Informatics School (AIS), Universiti Teknologi Malaysia, KL Campus, Malaysia

{sharifi1400,masarat\_ayat}@yahoo.com {m-wkhair,shamsul,suhaimiibrahim}@utm.my

**Abstract** - One major problem is to record software assets and update them in the organizations. Information Technology Service Management (ITSM) usually refers to ISO/IEC 20000 standard that the Configuration Management (CM) process known as heart of it. Configuration Management Data Base (CMDB) is part of the CM that records and updates changes and retrieves the services. This paper focus on propose a novel ITSM-based implementation method to maintain software assets in order to sustain organizations activities.

**Keywords:** ITSM, Software asset maintenance, CMDB, Configuration management

### I. INTRODUCTION

Maintenance is Actions necessary for retaining or restoring an equipment, machine, or system to the specified operable condition to achieve its maximum useful life. It includes corrective maintenance and preventive maintenance [1]. Corrective Maintenance refers to any maintenance activity which is required to correct a failure that has occurred or is in the process of occurring. This activity may consist of repair, restoration or replacement of components. Preventive maintenance means prevent of the equipments failure before it actually happened.

One kind of maintenance is Software maintenance. Software maintenance is defined in the IEEE Standard (IEEE 1219) as: "Software maintenance is the process of modifying a software system or component after delivery to correct faults, improve performances or other attributes, or adapt to a changed environment." [5] So software maintenance is corrective maintenance.

Software maintenance often define as a very broad activity that is include all work made on a software system after it becomes operational [2]. This covers the corrective- errors correction, the enhancement, deletion and addition of capabilities, adaptive – changes in the software environments and data requirements, the improvement of performance, usability, Perfective –new user requirements or any other quality attribute [3][4].

The importance of software maintenance is because it expends a large part of the overall lifecycle costs .

Moreover, Also business opportunities will be lost if there is no ability to change software quickly and reliably [6].

Software Asset Maintenance (SAM) is a sub set of software maintenance. SAM is defined as all of the infrastructure and processes necessary for the effective management, control and protection of the software asset throughout all stages of their lifecycle [7,8]. The goals of SAM are to reduce Information Technology (IT) costs and limit business and legal risk related to the ownership and use of software, while maximizing IT responsiveness and end-user productivity [9,10].

On the other hand, Information Technology Service Management (ITSM) refers to managing information system. It include ISO/IEC 20000, Information Technology Infrastructure Library (ITIL) and Capability Maturity Model Integration for service (CMMI- SVC).

The ITIL is a globally recognized collection of best practices for ITSM. ITIL is a framework of best practices to achieve quality service and overcome difficulties associated with the growth of IT systems. ITIL is organized into "sets" of books which are defined by related functions: service strategy, service design, managerial, service transition, service operation and continual service improvement [11].

ISO/IEC 20000 is the world's first standard for IT service management. The standard specifies a set of inter-related management processes, and is based heavily upon the ITIL framework [12].

Capability Maturity Model Integration for service (CMMI- SVC) is consists of the best practices that address development and maintenance activities that

cover the product life cycle from conception through delivery and maintenance and it can be used to guide process improvement across a project, a division or an entire organization[13].

In this paper at first ISO/IEC 20000 as the most accepted standard by companies will be described. Then the heart of this standard which is configuration management and Configuration Management Data Base (CMDB) as a repository will be introduced and at the end of this paper the applicability of CMDB to record , update and extract of software items will be discussed.

## II. ISO/IEC 20000 DESCRIPTION

ISO/IEC 20000 is the international standard [14] that describes the requirements for an ITSM system. The standard was developed to mirror the best practices described within the ITIL framework. ISO/IEC 20000 also supports other frameworks, such as Microsoft's Operations Framework [15]. ISO/IEC 20000 is comprised of two parts: a specification for IT Service Management (ISO 20000-1) and a code of practice for service management (ISO 20000-2) [16]. The Figure 1 shows the view of this standard.

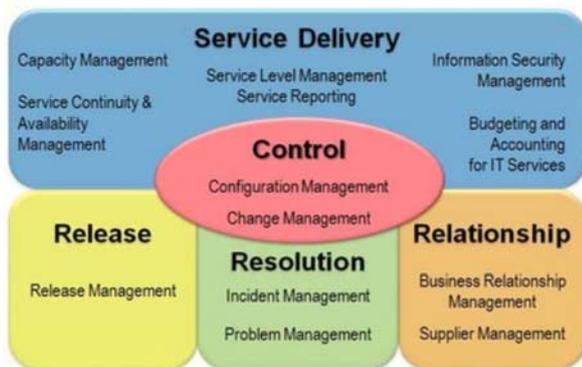


Figure 1: A complete view of ISO/IEC 20000[16]

ISO/IEC 20000 is in fact based upon an original pair of documents, BS15000-1/2[15]. It sets out the main processes to deliver IT services effectively [17] and it includes the following group processes:

### A. Service Delivery Processes

This group includes these processes:

**Capacity management:** This process is responsible to provide the required capacity for data processing and storage, at the right time and in a cost effective way and aligned the current and future requirements of the business.

**Service level management:** SLM is a written agreement between a service provider and customer that documents agreed service levels for a service.

**Information security management:** The purpose of this process is to check the compliance of IT system with the security policy as well ensuring that the stated level of security that is documented in the service level agreement for the stated IT services is done.

**Service continuity and availability management:** This process is responsible for defining, analyzing, planning, measuring and improving all aspects of the availability of IT services. Availability management is responsible for ensuring that all IT infrastructures, processes, tools, roles and ... are appropriate for the agreed service level targets for availability.

**Budgeting and accounting for IT services:** This process is responsible for accounting the costs of providing IT service and for any aspects of recovering these costs from the customers.

**Service reporting:** The main aim of this process is to produce agreed, timely, reliable, accurate reports for informed decision making and effective communication.

### B. Relationship Processes

This process is the interfaces between the service provider, customers and suppliers that are include the flowing process:

**Business relationship management:** The aim of this process is to establish and maintain a good relationship between the service provider and the customer based on understanding the customer and their business drivers.

**Supplier management:** The objective of this process is to manage suppliers to ensure the provision of seamless, quality services.

### C. Resolution Processes

The following processes are inside this process group:

**Incident management:** This process is responsible for managing the lifecycle of all incidents. The main objective of incident management is to return the IT service to customers as quickly as possible.

**Problem management:** The purpose of this process is managing the lifecycle of all problems. The primary

objectives of problem management are to prevent incidents from happening, and to minimize the impact of incidents that cannot be prevented.

#### *D. Control Processes*

This process group includes the following processes:

- Configuration management: Configuration management (CM) is the detailed recording and updating of information that describes an enterprise's computer systems and networks, including all hardware and software components. Such information typically includes the versions and updates that have been applied to installed software packages and the locations and network addresses of hardware devices. All this information records in and retrieves from CMDB that it will be discussed later [18].

- Change management: This process is responsible for standard methods and procedures that are used, such that changes can be dealt with quickly, with the lowest possible impact on service quality

#### *E. Release Processes*

The role of this process is planning, scheduling and controlling the movement of releases to test and live environments. The primary objective of release management is to ensure that the integrity of the live environment is protected and that the correct components are released.

### III. SOFTWARE ASSET MANAGEMENT (SAM)

It is clear that software is an important intangible asset which acts as a vehicle for value creation in businesses today. Based on a survey in Europe and US[19], more than 70% of them said that their core software assets were critical or very critical for their business strategy. Industry experts estimate that the amount of software in organizations is doubling every five to seven years. Software assets record and retrieve knowledge about business processes, about internal and external collaborations in the core of business and in relation with business partners. So, software is a vital intangible asset and it need to be managed.

#### *A. Problems and Challenges in Managing the Software Assets*

Despite the overwhelming dependence of organizations on their software assets, these organizations do not have enough information on their software assets and it makes trouble for them.

Organizations are under the pressures of managing software systems which are bigger and more complex than those from past years, but also need to meet increasing demands for higher quality to meet organization's objectives.

One important problem is the most organizations do not know how much software is running in their organizations. One research shows 60% of the surveyed companies not know the size of their core software assets [20]. The problem gets exacerbated by the fact that software is not visible and has a tendency to live for ever. As it can be seen, one major problem is software asset counting in the organizations. Due to this problem CMDB can be proposed to solve it. There are some different CMDB implementation approaches which will discuss in the following part in more details.

#### *B. Current Software Asset Techniques*

Usually, the CFOs consider the software assets like other hereditariness staff in the organization. However, it makes some difficulties for them. It is difficult to:

- Track and trace several versions of one tool
- Record every change of a specific tool
- Document several licensees of every tool and their places
- Tool owners due to network users
- Shared responsibilities and limitations

So, it needs a modern system to handle all related concerns and the best option is to employ a CMDB-based tool to manage them.

### IV. CONFIGURATION MANAGEMENT DATA BASE

ITSM, as it claimed above, includes several processes with CM as a core within this group. CM is a process that keeps all required information about services, service components, relationships, and other items accurate and up to date.

CMDB contains all relevant information about the components of the information system used in an IT organization services and the relationships between those components [21]. A CMDB provides an organized view of data and a means of examining data from different perspectives. In this context, components of information system are referred to as Configuration Items (CI). A CI can be any conceivable IT component, including software, hardware, document and personnel as well as any combination of them. In this paper, the main focus is on software asset and their relation. So, CI refers to software asset of the organization.

A. CMDB Implementation Approaches

As already stated, CMDB is a database that contains all relevant information about the software components of the information system used in an IT organization service that supports business processes of the organization. It can be said that CMDB should be able to answer these questions:

- What do we have?
- Where is it?
- What state is it in?
- Who is responsible for maintaining it?
- Who uses it?
- What depends on it?
- What does it depend on?
- What is it part of?

The challenges and risks are associated with the actual implementation of CMDB. However it specifically depends more on the approach taken for the project. There are four approaches to implement CMDB:

• Top-down approach

Top-down approach means that the analyzer should start by identifying its most critical softwares and then concentrate on defining only the key performance enablers for the delivery of those services.

• Bottom-up approach

This approach involves cataloguing every device and application plugged into infrastructure. This process starts with collecting small software assets and then merging them in upper level till the route which is software.

• Iterative approach

It is an initial loose strategy firmed up via gaining experience.

• Ad-hoc approach

This approach involves making decisions explicitly on project basis [22].

However, the most applicable and usable approaches are top down and bottom up approaches. The planning considerations and challenges of these approaches are shown in Table 1.

Every approach has its own strengths and weaknesses. Basically, it depends on some reasons e.g. experience of consultants and type of project. Implementation of CMDB begins by selecting a suitable approach. Respective approach will record its relation, changes and case histories.

B. The best CMDB Implementation Approach for Software Maintenance

As it already stated there are four different approaches which are Top-down, Bottom-up, Iterative and Ad-hoc approaches to implement CMDB in the

organizations that the most commons of them are Top-down, Bottom-up approaches. Implementation process of Top-down approach in the CMDB, essentially starts with breaking down the whole assets to gain insight into its compositional sub-sets. So, at first, an overview of the assts will be formulated and specificities but not any detailing about first-level subsets. This process continues until achieve base elements. The tracking in this approach is like a tree-tracking. Figure 2 shows a sample of this approach.

Implementation process of bottom-up approach tries together sub-sets to give rise to grander sets, thus making the original set. In a bottom-up approach the individual base items of the assets are first specified in great detail. These items are then linked together to form larger subsets, until a complete top-level set is formed. Figure 3 shows a sample of this approach.

Table1: Comparison challenges of two Approaches

	planning	challenges
Top down approach	<ul style="list-style-type: none"> <li>• Objectives and benefits</li> <li>• Organizational readiness, teams and responsibilities</li> <li>• Design and scoping</li> <li>• Metrics for success</li> </ul>	<ul style="list-style-type: none"> <li>• General understanding of what CMDB is and what it offers</li> <li>• How and where to apply it</li> <li>• Securing the support of senior management</li> </ul>
Bottom up approach	<ul style="list-style-type: none"> <li>• Granularity of CIs (if choosing all approaches may lead to potentially risk)</li> </ul>	<ul style="list-style-type: none"> <li>• General understanding of CMDB concept and its offer</li> <li>• Enabling technologies</li> <li>• Staff efforts</li> </ul>

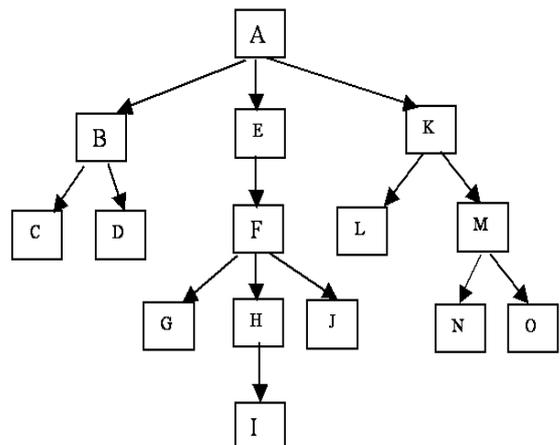


Figure 2: A sample view Top-down approach

Every one of these approaches has some advantages and disadvantages. Some weaknesses of bottom-up approach are as following[23,24]:

Firstly, in this approach, CMDB should be implemented with a clear granularity of assets with clear details that at the end may not use at all. One example is to put some options for a server that it does not need any software application. So, in this case, some parts of Data Base (DB) table will be empty and at the end it made a problem for DB administrators to manage such mostly empty and huge DB. Secondly, some of the softwares are embedded in the systems. Managing and counting such software assets is very difficult. Some times the granularity has not any meaning for them because they are not independent asset. Thirdly, due to nature of this approach, so many staff should be conducted and it needs to clarify related concepts to them. Fourthly, this approach is too time- consuming. Lastly, the cost of this approach is high due to high demand of hardware capacity. But the advantages of this approach are included:

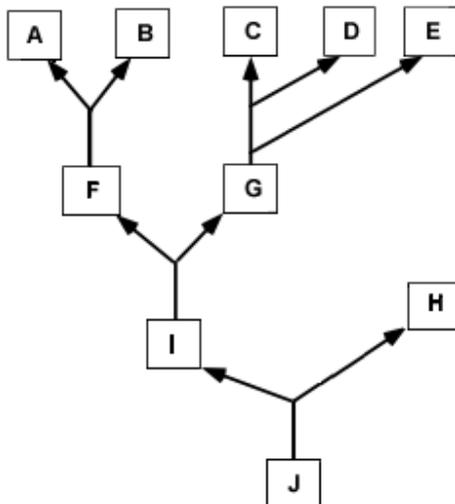


Figure 3: A sample view Bottom-up approach

At first, there is no need to restructure or redesign CMDB. Second, the founded DB is completed and enough comprehensive. Third, there is no redundancy to put so many same components and asset specifications. And the last there is no chance to leave any asset at the end of work.

On the other hand, the weaknesses of Top-down approach are that the completed CMDB using this approach becomes more complex than other approaches. Also, like ad-hoc approach it is too vital to select starting point of this approach. But, this approach has more

advantages for software assessment compare to Bottom-up approach. At first, this process starts with clarification of software categories and type of the system. Then, it narrows every one of sub sets in more details. So, final CMDB will be smaller than previous approach. Second, it is much less time consuming compare to previous approach due to very optimized way of processing design based on needs and current available assets. Third, the approach needs fewer costs to implement. In addition, the structure of DB will be designed, clearly and suitable for account current assets. Instead, in the top-down approach, it can not be exactly said that how many level should be prepared to finish the process. The summary of strengths and weaknesses of these approaches has been shown in Table 2.

So, based on the explained reasons and their comparison, the best approach to implement CMDB that can be proposed is top-down approach.

### V. VALIDATION OF PROPOSED METHOD

CMDB is using in all ITSM based tools to handle service incidents and related activities in the organizations [24]. It can be said that the main infrastructure of all tools are this Data Base which has been embedded to all of them, especially Remedy from BMC software development company [24]. Hence, this method has been validated in service maintenance and same method can be applied in software maintenance as well.

Table 2: The strengths and weaknesses of top-down and bottom-up approach [23]

Approach	Strengths	Weaknesses
Top-down	<ul style="list-style-type: none"> <li>• tree tracking approach</li> <li>• less price</li> <li>• easy to use</li> <li>• small size</li> <li>• much less time consuming</li> </ul>	<ul style="list-style-type: none"> <li>• complex</li> </ul>
Bottom-up	<ul style="list-style-type: none"> <li>• no need to restructure or redesign CMDB</li> <li>• complete and comprehensive DB</li> <li>• no redundancy</li> </ul>	<ul style="list-style-type: none"> <li>• not clear granularity of assets</li> <li>• difficult to managing and counting</li> <li>• high cost</li> <li>• big size</li> </ul>

## VI. RESULTS

Proposed method can play a very beneficial and effective role in software maintenance like what it is doing in service maintenance right now. This method can optimize and accurate related activities to have up-date information all the times. Consequently, this method is one of the best solutions which can be accepted and applied by CFOs and CIOs in the organizations.

## VII. CONCLUSION AND FUTURE WORK

Software asset maintenance is becoming an important consideration due to huge numbers of different software and embedded tools in the businesses. Managers of organizations face to some challenges on software asset maintenance which one of the most important of them is software tracking. Also, ITSM is a framework which tries to improve quality of services in the organizations. The core of this framework is CM that the main aim of it is to record, update and retrieve the IT assets and there are four different approaches to implement this database in the organizations. This paper conveyed this message that this DB can be used in SAM to record software assets and their affiliations to other softwares and equipments. It proposed that the best approach to implement CMDB in the organizations is Top-down approach due to nature of software assets and it brings the advantages like less price, clear structure and easy to use for DB administrators.

One Major consideration is to evaluate current developed tools to decide which one is the best in terms of ITSM support. However, there are companies which advertize their tools is supporting ITSM approach, but their claim needs to be evaluated and examined to decide whether they really support that or not.

## ACKNOWLEDGMENT

This research is supported by the Ministry of Science and Technology and Innovation (MOSTI), Malaysia and Universiti Teknologi Malaysia (AIS School).

## REFERENCES

- [1] Business dictionary, access: Jun 2009, <<http://www.businessdictionary.com/definition/maintenance.html>>
- [2] Martin, J., Mc Clure, C., "Software Maintenance – the Problem and its Solutions", Prentice Hall, Englewood Cliffs, NJ, 1983.
- [3] Kagan Erdil Emily Finn Kevin Keating Jay Meattle Sunyoung Park Deborah Yoon , "Software Maintenance As Part of the Software Life Cycle Comp180: Software Engineering" Prof. Stafford December 16, 2003
- [4] Gerardo Canfora and Aniello Cimitile "Software Maintenance" University of Sannio, Faculty of Engineering at Benevento Palazzo Bosco Lucarelli, Piazza Roma 82100, Benevento Italy 29 November, 2000
- [5] IEEE Std. 610.12, "Standard Glossary of Software Engineering Terminology", IEEE ,Computer Society Press, Los Alamitos, CA, 1990.
- [6] K. H. Bennett , V.T Rajlich "Software Maintenance and Evolution: a Roadmap "Research Institute for Software Evolution University of Durham UK , Department of Computer Science Wayne State University Detroit, MI 48202, USA
- [7] Kelway "Software Asset Management", Kelway (UK) Ltd is registered in England and Wales, company number: 2465350 Registered Address: Building 1000, Dockside Road, London, E16 2QU.< <http://www.kelway.co.uk/content/SAM> >
- [8] "Information technology — Software asset management-- Part 1: Processes", International Standard, International Organization for Standardization and International Electrotechnical Commission, 2006-05-01, pp. 5
- [9] a b c "Information technology — Software asset management-- Part 1: Processes", International Standard, International Organization for Standardization and International Electrotechnical Commission, 2006-05-01, pp. 5
- [10] Florin Dumitriu, Gabriela Mesnita, "Issues of Communication and Cooperation in Offshore Software Development " , publisher: social science research network, May 2006
- [11] Bon J. V.et. al. (2007). "Best Practices: Introduction to ITIL", TSO Publications, Norwich, June 2007.
- [12] The World Wide Web Journal, The ITSM Journal: ISO 20000, Posted by the Editor: 2007 <<http://www.w3journal.com/7/s3.rifkin.wrap.html> >, access: Feb. 2009
- [13] "The Importance of Qualityc" , procise gmbh Hanauer LandstraÙe 293B 60314 Frankfurt, access :6 March 2009 , <<http://www.procise.com/methods/itil/iso-20000/>>
- [14] ITIL SERVICE MANAGEMENT , ISO/IEC 20000 essentials , <http://itservicemngmt.blogspot.com/2008/02/isoiec-20000-essentials.html> , FEB 22, 2008.
- [15] Dugmore, Jenny (2006). Achieving ISO/IEC 20000 – "The Differences Between BS 15000 and ISO/IEC 20000". BSI Standards. p. 124. ISBN 0580473481.
- [16] ARNOST KATOLICKY, " ITIL - ITSM – COBIT", <<http://www.akamonitor.cz/ITIL/2006/01/iso-20000-1-iso-20000-2.htm>> ,Location: Plzeñ : Czech Republic , Jun 2006
- [17] "A Library of Resources for ISO 20000 Service Delivery", <<http://www.20000-toolkit.com/sd-library.htm>> access: Jan 2009
- [18] ITIL Glossary of Terms, access: Dec 2008 <[www.cmpp.net/CMS/Media/Docs/ITIL/ITIL%20Glossary%20of%20Terms01.doc](http://www.cmpp.net/CMS/Media/Docs/ITIL/ITIL%20Glossary%20of%20Terms01.doc)>
- [19] "Knowledge Transfer Suite 22827W093 Geneva Road Winfield, Illinois USA 60190, 2008 <[http://www.knowledgetransfer.net/dictionary/ITIL/en/Availability\\_Management.htm](http://www.knowledgetransfer.net/dictionary/ITIL/en/Availability_Management.htm)> access: 18 Dec 2008
- [20] "World's First Business Organization to Achieve ISO 20000-1 Certification " , publisher: Digital Forum <[http://www.digitalforum.accenture.com/DigitalForum/Global/ViewByTopic/eAI/0607\\_First\\_Bus\\_Org\\_ISO.htm](http://www.digitalforum.accenture.com/DigitalForum/Global/ViewByTopic/eAI/0607_First_Bus_Org_ISO.htm) > Jun 2006
- [21] "Configuration Management " ,publisher :TechTarget, Location TechTarget 117 Kendrick Street, Suite 800 Needham, MA 02494, access May 2009 <<http://www.bitpipe.com/tlist/Configuration-Management.html>>
- [22] Sharifi, M.; Ayat, M.; Sahibudin, S. "Implementing ITIL-Based CMDB in the Organizations to Minimize or Remove Service Quality Gaps", Publisher: Modeling & Simulation, 2008. AICMS 08. Second Asia International Conference on, Volume , Issue , 13-15 May 2008 Page(s):734 – 737 ,IEEE, 2008.
- [23] Sharifi, M.; Ayat, M.; Sahibudin, S.; Ibrahim,S.; "The Most Applicable CMDB Implementation Approach for SITUs and SMEs Companies", Publisher: 12th IBIMA Conference 29 – 30 June 2009
- [24] Sharifi, M.; Ayat, M.; Sahibudin, S., "Combining ITIL, COBIT and ISO/IEC 27002 in Order to Make a Comprehensive IT

Framework in Organizations”, Publisher: AMS ,Location: IEEE  
,2008  
**mics Ltd.**

[25] Atherton M.( June 2007),Deploying CMDB Technology  
Pragmatism and realism will deliver the benefits,Freeform Dyna