

Deriving Software Acquisition Process from Maturity Models—An Experience Report

Hussain Alfaraj, Shaowen Qin

School of Computer Science, Engineering and Mathematics, Flinders University, Adelaide, Australia.
Email: hussain.alfaraj@csem.flinders.edu.au

Received October 26th, 2009; revised December 15th, 2009; accepted December 20th, 2009.

ABSTRACT

The establishment of an existing practice scenario was an essential component in providing a basis for further research in the area of COTS software acquisition within the organisation. This report details the identification of means of describing the existing practice of software acquisition within an organisation and identification of models that could be used to present this view. The chosen best practices descriptions for the idealized model were maturity models, including SA-CMM, CMMI-ACQ, and ISO/IEC 12207. This report describes these models briefly and then describes the process of identifying the requirements for idealizing these maturity models into process frameworks that could be identified to actually business process models from a real organisation in order to identify gaps and optimizations within the organisation's realization of the best practices model. It also identified the next steps in identification of the theoretical best practice framework, which will involve translation of the model to YAWL Petri nets and simulation of the process in order to identify potential modelling flaws or issues with framework efficiency. Implications of the currently ongoing research include the identification and correspondence of specific tasks and activities from ITIL and CoBiT frameworks with the generic key process areas of software acquisition frameworks and identification of sufficiently detailed structural framework models for each level in order to identify appropriate frameworks for application even in cases where these frameworks were not explicitly identified by the organisation or the researcher.

Keywords: BPM, Workflow, Software Acquisition, Simulation, CoBiT, ITIL

1. Introduction

The researcher's current area of focus is on the derivation of the software acquisition model in use within organisations for commercial off the shelf (COTS) software acquisition using BPM tools. Two different areas of investigation were chosen for this analysis, including determination of an idealized model based on current best practices and the description of the actual practice within the organisation under study. While many organisations do attempt to undertake the development of processes under the best practices frameworks described below, many organisations do not succeed in this goal either due to deliberate divergence from the best practice in order to accommodate organisational realities or because of inability to reach the best practices condition for another reason [1].

Existing best practice models for software acquisition are built on commonly accepted standards that either represent software acquisition as a standalone process or integrate the acquisition process into the software lifecycle. The diversity of these models means that an organi-

sation is likely to be able to choose an appropriate model for its needs, but none of the best practices models is likely to be fully adequate. Valuable information can be gained by comparing the process as enacted within the organisation with the template provided by the best practices model. This research required building theoretical models for three commonly used software lifecycle standards, the Capability Maturity Model for Software Acquisition (CMMI-ACQ), the Software Acquisition Capability Maturity Model (SA-CMM), and the ISO/IEEE 12207 software lifecycle standard (which integrates software acquisition into the lifecycle model as compared to the other two models, which describe it separately).

This report describes these models in brief and then focuses on the process of describing the idealized template for the oldest model, SA-CMM, in order to demonstrate the engagement of analytical tools. A discussion is also provided regarding the next steps in identification of organisational optimization and matches to this process. The importance of this report is that it provides a scientific understanding of the models as they relate to the software acquisition process. Organizations that have likely used

other methods in the past that may not have taken fully into account both the managerial issues and the tasks related to the best outcomes of software acquisition will be provided not only with a model for the process, but an understanding of the important elements of the model and how to integrate them into real-world application.

2. Best Practices and Standards

Three best practices models or standards were identified for inclusion based on the completeness of the standard and the level of actual use within these organisations. SA-CMM, the oldest model that is examined, is still in active use within some organisations, while others have enacted its successor, CMMI-ACQ [2]. Both of these models were developed by the Carnegie Mellon University Software Engineering Institute to support the development of processes in the organisation. The third model ISO/IEC 12207, describes an overall view of the software lifecycle that includes the acquisition process. Each of these models presents a foundation that is important to understand in order to examine the develop a model of the software acquisition process that integrates both management issues and key tasks and procedures that are to be followed for the opportunity for the best outcomes for an organization. The examination not only examines these theories, but also helps to bridge theory with real-world needs and concerns.

2.1 SA-CMM

The SA-CMM best practices standard (currently version 1.03, released 2002), was developed to provide a capability maturity model that could be used in the context of software acquisition [3]. Although it was developed for use by the United States Department of Defence, it has been widely used in educational and industrial contexts as well. Companies have found that the SA-CMM best practices standard provided a straight-forward process by which to understand the important issues related to software acquisition [3]. The SA-CMM model, which is shown in **Table 1**, is based on a five-layer model in which each level delimitates a different level of maturity. The maturity level is made up of key process areas, which include goals, institutionalization features (commitment to perform, ability to perform, measurement and analysis, and verification), as well as activities [3]. The SA-CMM model is a staged model, indicating that the results are cumulative—for example, it is necessary to meet the requirements of Level 2 in order to achieve Level 3. Without completing one stage successfully before moving to the next, important issues are not entirely handled, and the result is likely to not be the best outcome for an organization [3].

The goal of the SA-CMM best practices model is to provide a description of the software acquisition process that can be adapted to any organisational context, and as

Table 1. SA-CMM model

Level	Focus	Key Process Areas
1. Initial		Competent People and Heroics
		- Transition to Support - Evaluation - Contract Tracking and Oversight
2. Repeatable	Basic Project Management	- Project Management - Requirements Development and Management - Solicitation - Software Acquisition Planning
		- Training Program Management - Acquisition Risk Management - Contract Performance Management
3. Defined	Process Standardization	- Project Performance Management - User Requirements - Process Definition and Maintenance
		- Quantitative Acquisitions Management - Quantitative Process Management
4. Quantitative	Quantitative Management	- Acquisition Innovation Management - Continuous Process Improvement
5. Optimizing	Continuous Process Improvement	

such it is a very wide-ranging [4]. However, this characteristic makes the SA-CMM difficult to implement within an organisation, as specific requirements for this implementation in terms of tools or identified techniques are not complete. Another significant gap in the SA-CMM is that it does not specify requirements, which is considered to be essential for determination of the appropriate fit between software and organisation [5]. As such, although the SA-CMM is used in many organisational contexts it may not be appropriate or optimal for all organisations.

2.2 CMMI-ACQ

The CMMI-ACQ best practices description (current release 1.02, September 2007) is the successor to the SA-CMM model (although the SA-CMM model is still in use in many organisations). [6] This model is built on the older SA-CMM description, but provides a considerable improvement over this model. It was generated from the CMMI Architecture and Framework, an existing model that describes various aspects of lifecycle development of software [7]. This model is also intended to be a generic model for organisational process description and improvement, applicable to any organisation [7].

Unlike the SA-CMM model, there are three levels of inclusion for CMMI model components, including required, expected, and informative components [7]. There are 22 identified process areas within the CMMI-ACQ model, and like SA-CMM, the CMMI-ACQ model can be used as a staged model; however, unlike the SA-CMM model, the CMMI-ACQ model can also be used in a con-

tinuous representation, which allows for transition between stages depending on existing capabilities [7]. What this means for an organization is that it does not have to feel that each stage must be completed, even if it does not entirely apply to the organization or its needs, before moving to the next. An organization has some control over the model in the ability to make changes or adjustments to specific issues or concerns in relation to how it operates and the needs that it considers to be important.

Under the staged model, in which, as with SA-CMM, transitions occur from one model to the next sequentially and determined by achieving core competencies from previous stages, there are five different stages that the organisation can move through (Initial, Managed, Defined, Quantitatively Managed, and Optimizing); these five stages roughly correspond to the five stages of the SA-CMM model [7]. However, the continuous model allows for a sixth stage, Incomplete (in effect a Level 0). This extra level in the model can be important for an organization because the assumption that a level has been completed is not possible. An organization can indeed be incomplete with regards to its acquisition efforts. The difference between these two models is that the standard presents a capability model in the continuous representation (which is intended to describe the individual capabilities of the organisation at whatever level they have occurred), while the staged representation is intended as a maturity model, representing the overall maturity level of the process within the organisation [7]. This means that the staged and continuous representations do have different approaches to tasks and requirements for competence attainment, but the models are largely consistent with each other.

While the CMMI-ACQ description did rectify some of the challenges of SA-CMM including development of a requirements determination activity area and introduction of specific predefined tasks, it does retain some challenges as well. These include lack of guidance offered on the priority of process areas within the continuous representation [8]. This lack of prioritization means it can be challenging for implementers of the process to determine appropriate priorities or task ordering; while decision support models have been identified that can rectify this challenge to some extent it remains one of the highest barriers to organisational implementation of this process. In addition, for an organization that may be beginning the process of truly thinking about best practices related to software acquisition or having a scientific theory to guide software acquisition, the use of a broad model that is more oriented toward ideas as opposed to specific tasks can be challenging. An organization can become so concerned about ideas and concepts than it ignores the tasks that need to be completed in order to successfully complete software acquisition and ensure that it meets the needs of the organization once it has been completed.

2.3 ISO/IEC 12207

The third standard that was undertaken during this analysis process was ISO/IEC 12207, which implements the software acquisition process as part of a full description of the software lifecycle. ISO/IEC 12207 was the first standard to describe the full software lifecycle [9]. The ISO/IEC 12207 family of standards is one of the most commonly used standards for the definition of the software lifecycle process as a whole, including the software acquisition process, which is embedded in the standard [10]. Starting from the acquisition process, the ISO/IEC 12207 standard describes the full software lifecycle, including aspects such as human resources management and infrastructure life cycle management. This means that specific tasks that need to be completed, as opposed to only focusing on issues and concepts, are part of the model. In essence, this model allows for a bridge to be created between purely issue-related software acquisition and the tasks and duties that are part of the software acquisition process [9]. However, only the software acquisition process model was considered to be relevant for this research process.

Unlike SA-CMM and CMMI-ACQ, the ISO/IEC 12207 model is not a capability or maturity model *per se*, but is instead a lifecycle model. Also unlike the previous two models discussed, it clearly defines operations, activities, tasks, and provides a complementary Supply Process that outlines the operations and activities required of the supplier of the software [9]. This view of the process from the supplier's viewpoint increases the potential that requirements of the organisation acquiring the software are met, because the Supply Process specifically addresses the requirement to meet purchaser requirements [9]. Also unlike the CMMI-ACQ and SA-CMM model, the ISO/IEC 12207 standard is not a staged model that is based on the stage of organisational maturity, but is instead a single process designed for use at all levels of maturity [9]. This does allow the organisation to build competence in these processes over time, but does not provide a means of determining the organisation's maturity. However, there are sub-processes offered that allow for identification of greater detail in the process if required. The model is much more detailed in this regard, which is important for an organization that may be undertaking software acquisition efforts in a scientific manner for the first time. The specific details that are listed provide a better guide for what needs to be performed in order to achieve a successful outcome.

Thus, while the ISO/IEC 12207 model is not appropriate for determining the maturity or capability level of an organisation it does allow for the development of specific skills and competencies related to software acquisition by spelling out the required process for effective software acquisition (and through the complementary Supply Pro-

cess, the effective supply of software to organisations). As such, this can be seen to be a standard that would be put to a different use from the organisational requirement that led to the use of the capability maturity models described by the SA-CMM or CMMI-ACQ frameworks.

3. SA-CMM Conversion Process

The description above provides an overview of the best practices frameworks. However, there is still the question of how these textual descriptions can be defined as a framework that can be directly compared to process models derived from organisational studies. There were a number of issues identified in this experience. First, the textual descriptions offered little information regarding the specific tasks and activities. Second, the varying maturity levels within this capability maturity model offer different activities and processes, making the model complex to describe as a single model framework. In order to overcome these challenges, the specific tasks and activities were kept generic in order to comply with the framework of the discussion, and the individual models were described in separate frameworks. It is understood that since the SA-CMM model is a cumulative framework, that organisations would be able to compare their observed processes serially against the models in order to determine which capability level met the requirements most effectively.

3.1 Description of the Framework

The SA-CMM framework presented is the Level 2 approach to software acquisition. However, a similar process was followed to describe the process at all five levels of the organisation. (This process was also performed for all levels within the CMMI-ACQ framework as well as the Acquisition process within ISO/IEC 12207). The process was as follows:

- 1) Identify key process areas that were described within the model;
- 2) Create a definition of the key process area that textually described the inputs, process, and outcomes of the process area;
- 3) Identify inputs;
- 4) Identify outputs;
- 5) Identify people;
- 6) Identify cost.

The key process areas identified at Level 2 (Repeatable) included Software Acquisition Planning, Solicitation, Requirements Development, Project Management, Contract Tracking and Oversight, Evaluation, and Transition to Support [3]. As can be seen in **Figure 1**, the key process areas are a mixture of competencies and process areas, including technical, project planning and management, and legal aspects of the acquisition process, which indi-

cates that the process will engage different individuals and organisational competencies within the region.

The process flow indicated by the Level 2 (Repeatable) description with the SA-CMM model was exceptionally simple. This process was a linear process with little room for deviation from the existing model or translation from one end of the model to another.

The model above describes the process of software acquisition at maturity Level 2 within the SA-CMM. The steps involved in software acquisition in the model are standard in terms of key process areas, but actual tasks and activities vary depending on the organisation. The issues of cost and resource allocation are strongly dependent on individual implementation and are not determined in the standard; as such, they will need to be determined on observation of individual implementations of the standard. However, the key process areas must all be identified and effectively engaged in if the firm wishes to move beyond the Level 2 (Repeatable) level of implementation, just as it was necessary for all the key process areas at Level 1 to be met effectively in order to move to Level 2 [7]. Because of this, a maturing software organisation will move into an effective implementation of all identified key process areas before moving forward to the Level 3 (Defined) model. As discussed above, there is a considerable challenge within this model, as tasks and activities are not actually clearly defined and there is no way to identify a generic standard for tasks and activities; as such, there will be representative tasks and activities identified in order to attempt to create a framework that can be used to describe a generic situation that meets the demands of a Level 2 organisation.

3.2 Identification of Tasks and Specific Activities

The model above has an obvious weakness, in that it does not describe specific tasks and activities, but instead focuses on the identification of process and management related functions. Once again, the disconnect that seems to exist between purely theoretical concerns as opposed to concerns related to real-world software acquisition and implementation are noticeable. In practice, the organisation will need to implement the best practices framework with actual tasks and activities, which may be refined from organisational needs or may be identified from additional best practices frameworks. The best practices frameworks that have been identified as ideal for use in construction of the framework models include the Information Technology Infrastructure Library (ITIL) best practice guide, published by Great Britain's Office of Government Commerce, and the Control Objectives for Information and Related Technologies (CoBiT) framework, which provides IT governance best practices [11–13].

The ITIL framework, which is shown in **Figure 2**, is commonly used in conjunction with CoBiT to incorporate governance and best practices, and there is a specific guideline intended to facilitate this co-incorporation [11]. The applicable ITIL volume is the Software Asset Management volume, which addresses software management at all stages of the lifecycle [14]. These additional best practices frameworks mimic the addition of supplemental frameworks, policies and standards within an actual organisation in order to determine appropriate tasks and activities. Additionally, researchers have identified these frameworks as being commonly implemented within the organisational environment, meaning that it is likely that an organisational study will reveal a similar process of identification of actual tasks and activities that would take place. As such, this is considered to be an appropriate supplement for the structure demonstrated above.

The combining of the ITIL and CoBiT models can actually be performed with relative ease. **Figure 3** provides a model for combining ITIL and CoBiT. Combining the models overcomes the problem of the lack of specific tasks and activities that is found in the SA-CMM Level 2 process flow. The combined ITIL and CoBiT model brings together management functions and company activities. In essence, what has occurred in the combined model is that software acquisition as moved from being purely managerial in nature to something that involves employees at all levels of an organization.

4. Refinement of the Framework

This report describes only the first iteration of a process

that is expected to have several stages of refinement. One potential way in which the identified models can be refined and further clarified is the use of simulation to identify potential difficulties and challenges within the framework. Business process simulation is commonly used in organisational environments for such tools as business process re-engineering and new process implementation, because it allows for identification of flaws within the proposed process Model [15]. The use of simulation within this context allowed for the identification of areas that could be problematic if implemented in actual practice. These areas will then be analysed in order to determine whether this is a specification error or whether it represents an actual area of implementation difficulty or inefficiency within the best practices framework. In the first case, the model will be refined to account for the identified difficulty, while in the second case further research will be performed with this area of weakness as a focal point.

The identified simulation and modelling tool that will be used for this process is YAWL (Yet Another Workflow Language), which is an open source workflow modelling and simulation tool [16]. This tool has been used extensively in academic BPM research as it is extensible and has a greater level of flexibility than most commercial offerings, which are primarily intended for analysis and are not aimed toward researchers. This process will require translation from the current BPML to YAWL’s Petri nets structure, but the use of simulation in order to identify potential difficulties in the framework models that will be used for comparison of actual cases will provide signifi-

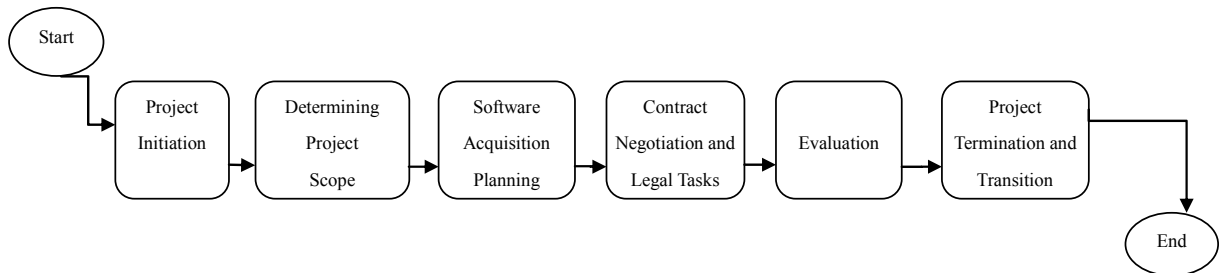


Figure 1. SA-CMM Level 2 process flow

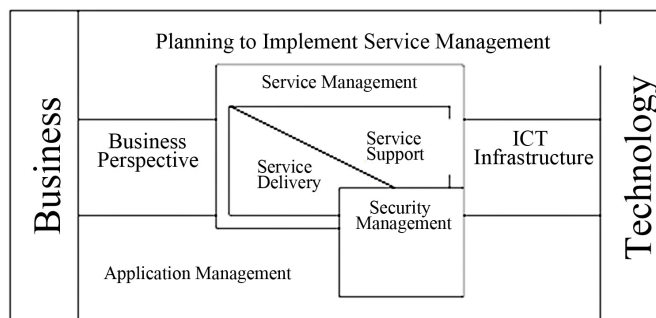


Figure 2. ITIL framework

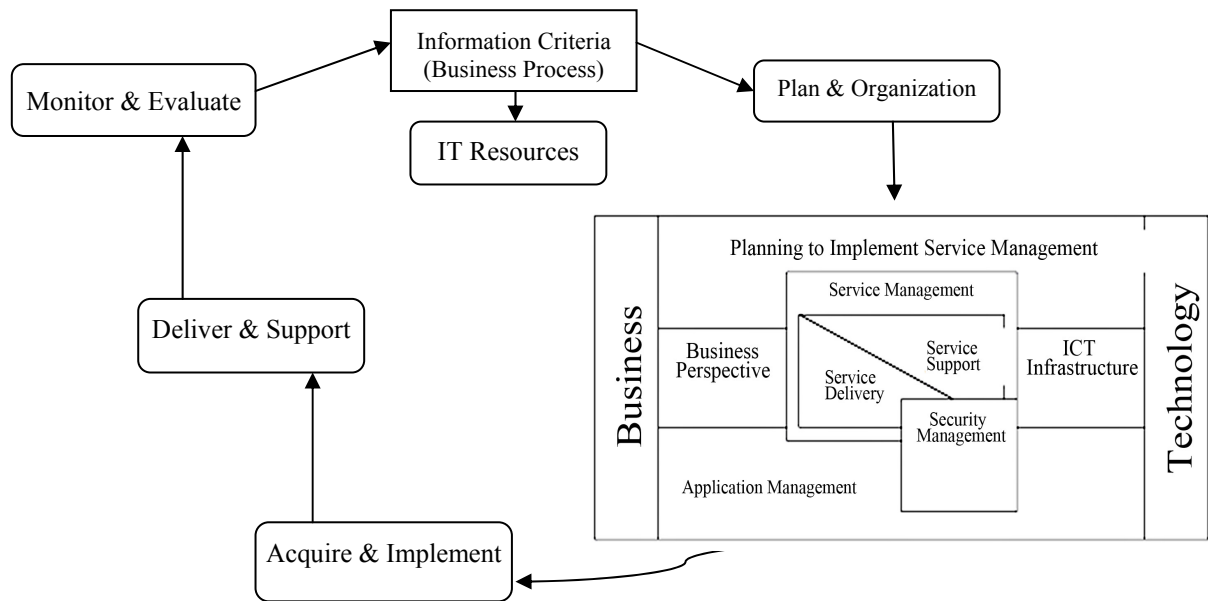


Figure 3. Combined ITIL/CoBiT model

cant benefits to the current research and as such this is considered to be an acceptable requirement.

5. Conclusions

The use of best practices frameworks and standards is common within organisations, but the specific needs of the organisation drives the choice of framework or standard. The three most common best practices standards provide different advantages and disadvantages to the organisation, and can be used in different ways to improve the organisation. The organisation that wants to engage in a specific process that is consistent across maturity levels and stages and includes specific tasks is likely to choose the ISO/IEC 12207 standard, while an organisation intent on developing capabilities in software acquisition will choose the CMMI-ACQ model. An organisation intending to develop maturity in the software process could use either CMMI-ACQ or the SA-CMM model. This discussion has demonstrated the process by which the textual descriptions were converted to idealized process frameworks that could be used to actualize process models identified from organisational studies. However, some issues have remained unresolved from this experiment, including the ability to identify specific activities and structures. This is not especially a problem with the ISO/IEC 12207 standard, which spells out specific organisational activities and a specific process, but does remain a challenge with SA-CMM and CMMI-ACQ, which are more flexible in terms of identification of the process activities and requirements (and in fact in some cases do not have specific requirements in this regard at all). In this discussion, it was suggested that the implementation of standards including ITIL and CoBiT could

be used to identify the specific tasks and processes that are missing from these structures. However, the use of organisational studies will be required in order to determine how the organisations themselves have resolved this issue—have these organisations used these best practices IT governance frameworks, or have they merged software capability and maturity models with uniquely identified models or practices? This is one of the outstanding issues that the researchers hope to resolve through the current research process.

It should be noted that the majority of organisations that use the SA-CMM framework for description of software acquisition processes currently operate at Level 2, which was the model described within this framework [17]. Thus, the description of the Level 2 framework was engaged in first in order to be able to describe the widest potential organisational pool. However, this process was followed for other frameworks and organisational levels as well. The researchers hope to use this generic description as a means of creating a template on which organisations can be matched following the identification of actual processes within the organisation in order to provide identification of processes from the research perspective, even in organisations that do not use the frameworks or models explicitly. This could yield information both for implementation of the models within other contexts (for example, providing information regarding optimizations in generic frameworks that can be identified from actual organisational studies) as well as provide information for organisations in terms of process improvement and increasing efficiency. Thus, the identification not only of key process areas, but also association of tasks and activities as described by ITIL and CoBiT to these key

process areas, is expected to be key to the eventual utility of this research.

REFERENCES

- [1] M. Biro, C. Deak, J. Ivanyos, and R. Messnarz, "From compliance to business success: Improving outsourcing service controls by adopting external regulatory requirements," *Software Process Improvement and Practice*, Vol. 11, pp. 239–249, 2006.
- [2] L. Anderson, M. Fisher, and J. Gross, "Case study: IRS business system modernization process improvement," Carnegie Mellon University, Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA, USA, 2004.
- [3] J. Cooper and M. Fisher, "Software Acquisition Capability Maturity Model (SA-CMM) Version 1.03," Technical Report, Carnegie Mellon University, Software Engineering Institute, Pittsburgh, PA, USA, 2002.
- [4] J. White, "Managing information in the public sector," M. E. Sharpe, London, UK, 2007.
- [5] J. A. Mykkanen, M. P. Tuomainen, "An evaluation and selection framework for interoperability standards," *Information and Software Technology*, Vol. 50, pp. 176–197, 2008.
- [6] F. Navarrete, P. Botella, X. Franch, "Reconciling agility and discipline in COTS selection processes," *Proceedings of the 6th International IEEE Conference on Commercial-off-the-Shelf [COTS]-Based Software Systems*, pp. 1–11. IEEE, 2007.
- [7] "CMMI product team: CMMI for acquisition, version 1.2: CMMI-ACQ, version 1.2," Technical Report, Carnegie Mellon University, Software Engineering Institute, Pittsburgh, PA, USA, 2007.
- [8] S. J. Huang, W. M. Han, "Selection priority of process areas based on CMMI continuous representation," *Information and Management*, Vol. 43, pp. 297–307, 2006.
- [9] J. Moore, T. Doran, A. Kark, "Systems and software engineering—software lifecycle processes," *Software & Systems Engineering Standards Committee of the IEEE Computer Society*, Institute of Electrical and Electronic Engineers, 2nd Edition, Piscataway, 2008.
- [10] Y. Hwang, J. G. Park, "Approaches and requirements to develop and improve the standard processes for a research and development organisation," *Systems Engineering*, Vol. 9, No. 1, pp. 35–44, 2006.
- [11] D. Nichols, "Governing ITIL with COBIT, DITY Newsletter," itSM Solutions LLC, Lexington, USA, 2007.
- [12] "ITGI/ISACA: COBIT 4.1," IT Governance Institute, USA, 2007.
- [13] "ITGI/OCG: Aligning COBIT, ITIL, and ISO 1799 for business benefit: Management summary," Office of Government Commerce, Norfolk, 2007.
- [14] "Office of government commerce: Software asset management," Stationery Office, London, 2003.
- [15] I. Lee, "Selected readings on information technology and business systems management," Idea Group Inc, London, 2008.
- [16] "YAWL: Yet another workflow language," <http://www.yawl-system.com/>.
- [17] C. Meyers and P. Oberndorf, "Managing software acquisition: Open systems and COTS products," Addison-Wesley, Sydney, 2001.