COTS Evaluation Supported By Knowledge Bases

Abdallah Mohamed[1] Tom Wanyama[1] Günther Ruhe[1] Armin Eberlein[2] Behrouz Far[1]

[1]University of Calgary, 2500 University Drive NW, Calgary, AB, T2N 1N4, Canada
{mohameda; wanyama; far}@enel.ucalgary.ca; ruhe@ucalgary.ca
[2]Computer Engineering Department, American University of Sharjah, UAE
eberlein@enel.ucalgary.ca

Abstract. Selection of Commercial-off-The-Shelf (COTS) software products is a knowledge-intensive process. In this paper, we show how knowledge bases can be used to facilitate the COTS selection process. We propose a conceptual model to support decision makers during the evaluation procedures. We then describe how this model is implemented using agent technologies supported by two knowledge bases (KB): the COTS KB and the methods KB. The model relies on group-decision making and facilitated stakeholder negotiations during the selection process. It employs hybrid techniques, such as Bayesian Belief Networks and Game Theory, to address different challenges throughout the process. In addition, the paper also describes how the COTS knowledge base can be used at three levels of usage: global (over the internet), limited (between limited number of organizations) and local (within a single organization).

1 Introduction

In order to improve the software processes and products, there is an increasing need for accumulating, managing, disseminating, and reusing knowledge all the way through the process of developing new software systems [11]. A good example for knowledge aggregation and reuse during software development can be found during the process of developing a COTS based system. Generally, COTS software components are ready-made software products that are acquired and used as-is. During the last decade, they have received a lot of industry attention due to their potential for time and effort saving [8]. However, developers are faced with the problem of ensuring that COTS products perform the functionality they claim to have at acceptable levels of quality, which is best described as the problem of getting the 'right' knowledge at the 'right' time. Several people in industry have stated clearly that if COTS products were selected improperly, the whole project would be more costly and time consuming. Therefore, selecting the ‘right’ COTS product requires a robust decision support model that considers different stakeholder opinions and other constraints, and that is supported by a well established knowledge-base that includes stakeholders' experiences as well as up-to-date relevant knowledge.

Generally, it is possible that different stakeholders’ preferences might be inconsistent and changing over time. Under such situations, there is a need for a COTS evaluation model that is flexible during aggregation of stakeholders’ opinions, and that allows evaluating different scenarios easily. The model should handle trade-offs

between different perspectives so as to obtain a single product that satisfies all stakeholders. This is in fact achieved through negotiations, traditionally carried out using the *consensus approach* [13]. Within this approach, stakeholders have to reach agreement on the evaluation criteria and their relative importance. Then, the developer team has to evaluate the functionality of different COTS candidates against the given criteria. Finally, the results for the evaluation process are aggregated in order to select the most convenient COTS. However, this approach has a lot of drawbacks. Firstly, it does not offer any means for trying out different scenarios to deal with conflicting stakeholders' preferences. Secondly, it does not deal with the situation where one (or more) stakeholder changes his/her preferences after the negotiation process without the involvement of other stakeholders. Thirdly, the uncertainty and interdependencies inherent in the evaluation process are usually not addressed. Fourthly, reusing information from previous evaluation processes is usually not considered.

In this paper, we propose a conceptual model for COTS evaluation that circumvents the above mentioned shortfalls of traditional models. We then describe how this model can be implemented by a system that uses agent technologies supported by two kinds of knowledge bases (KB) which are of key importance: the COTS Knowledge Base (CKB) and the Methods Knowledge base (MKB). The evaluation system simulates the evaluation process, allowing users to try out different scenarios and determine their impact on the evaluation outcome. The system also addresses changing requirements and stakeholders' preferences. In addition, our proposed system deals with the inherent uncertainty and requirements of interdependencies by using a bayesian belief network (BBN) to estimate the quality of each COTS candidate from different stakeholders' perspectives.

One of the challenges that we faced, while developing our approach, is to link our ideas and suggested procedures to the real-world. To deal with this point, we heavily studied the lessons learned from COTS-based software development repository provided by CeBASE (Center of Empirical Software Engineering)[2] and tried to address the stated issues. Mainly, CeBASE repository is a very rich source of real-world experiences and practical knowledge collected from different COTS-based projects over private industry and government. For example, it may contain hypothesis, good/bad practices, success stories or things to be avoided. The lessons-learned are submitted by different users and validated by on of the CeBASE experts.

Mainly, this paper is organized as follows: section 2 presents how the proposed system helps as a decision support system to address challenges inherent in the COTS selection process. Section 3 describes a conceptual model to be used during the evaluation process. In section 4, we show how this model can be implemented using an agent-based system supported by the mentioned types of knowledge bases, CKB and MKB. Moreover, we show the possible benefits gained from using this system. Finally, section 5 includes the conclusion and the suggestions for future work.

2 A Decision Support System for COTS Selection

A good Decision Support System (DSS) should be combined with firm knowledge modeling and management [12]. Moreover, it should be able to overcome many chal-

lenges inherent in the decision making process. In his paper [10], Ruhe has described these challenges for the process of COTS evaluation and selection. An ideal DSS should be able to address all these challenges perfectly. In our proposed system, we tried to cover as many as possible of those difficulties. Typical challenges are shown in Table 1 along with the techniques used in the proposed system to address them.

Table 1. The challenges of COTS selection process and how they are addressed.

Problem Characteristics		**Proposed Procedure**
Related to	**Description**	
Stakeholders	Typically, there are different stakeholders with different (and sometimes conflicting) views about the system to be developed.	- We employ the game theory along with agents assigned to stakeholders to facilitate a semi-automatic negotiation between them - We suggest using a knowledge base to provide necessary relevant information to different stakeholders which harmonizes their decisions
Dynamic Changes	System requirements are changing all the time due to stakeholders' continuous progressive understanding.	Using agents allows applying requirements changes faster to the selection process. It also can show any inconsistency that results from such changes and propagate them quickly to other stakeholders.
	Every 8-9 months, a new version for the COTS products is released with new (or modified) features.	Although it is difficult to keep track of all the changes in a real-time way, we suggest using a knowledge base that is updated continuously with the latest information regarding relevant COTS products to current domain.
Uncertainty	There is a large uncertainty about requirements, vendors, and the impact of COTS products on the system due to their "black box" nature	We tried to handle this issue as much as possible by using techniques that deals with uncertain and incomplete data, for example, we suggest using a bayesian belief network to estimate the quality of each COTS product. Nevertheless, we are still working on solving this problem more conveniently.
Constraints	There are several constraints that should be considered when choosing the best COTS product	In our approach, the negotiation process is controlled by some constraints such as the maximum cost or the underlying technology of the products. Nevertheless, some other constraints are more qualitative and uncertain such as the products' reliability and performance. We intend to cover this part in a future research.
Problem complexity	This problem is of high complexity due to the existence of many different COTS alternatives to select from.	Although the knowledge base should contain enough information about available COTS products relevant to a certain domain, we cannot claim that maintaining such knowledge base is easy. Sometimes, it takes more effort to maintain it than searching for end evaluating available COTS products in the market.
Objectives	When considering the selection problem, several incomparable objectives have to be considered. For example, the cost-benefit ratio, the reputation and trustworthy of the vendors, and the covered functionality.	In our proposed system, we have addressed only the covered functionality and the quality of the selected COTS product. However, we intend to extend our approach to cover all other objectives in a future research.

3 Conceptual Model for COTS Evaluation

We believe that COTS evaluation process is based upon the fact that relative importance of evaluation criteria depends upon the perspective of each individual stakeholder. For example, the System Design Team might be interested in the system needs and architectural issues, whereas the customers are more interested in the domain is-

sues related to their organizations and relative constraints, such as budget limits. We illustrate these differences in perspective in the proposed conceptual model shown in Figure 1. In the model, COTS alternatives are presented, along with relevant information, through a KB, which will be split later on into the COTS knowledge base (CKB) and the methods knowledge base (MKB). The KB gets its knowledge from different sources such as domain experts and vendors documents, as well as from previous evaluation processes. The *"update"* feedback loop aims at storing the results of previous evaluation processes to enable continuous learning, and hence, reduce the time and effort needed for future evaluation processes. Ideally, all the knowledge regarding COTS products functionality, quality, cost and other issues should be stored. However, this is very difficult to implement, at least in the current time. The COTS products are evaluated according to the needs and preferences of individual stakeholders, who are provided with feedback from the KB about the actually available COTS features. The evaluation process is carried out within the aggregation component, provided by the KB, by assessing COTS product features according to stakeholders' preferences. Different stakeholder preferences may lead to two situations. Firstly, all stakeholder preferences point to the same COTS product, and thus we don't have any problem selecting the product. The second situation is the one in which the stakeholders' preferences point at different COTS products, and thus a negotiation process is carried out, within the negotiation component under certain constraints. The goal of this negotiation process is to deliver the best ranking for the selected products, addressing as many as possible of the combined needs of all stakeholders.

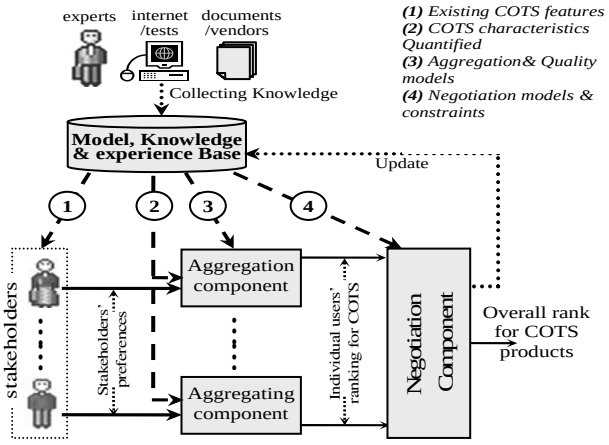

Figure 1. COTS evaluation conceptual model

4 Agent-Based System for COTS Evaluation

4.1 Why We Use Agents

COTS evaluation process has a number of characteristics that make it appropriate to use agent technology for the evaluation decision support system. These are:

- COTS evaluation is not a cleanly separable action, but is more permeative: it exists in multiple forms and at subtly different levels. For example, stakeholders make decisions on the following issues before the final evaluation process:
  1. Evaluation criteria that meet the needs of their respective organizations.
  2. Relative importance of the evaluation criteria.
  3. Support developing system in house or using COTS.

  This amounts to a distributed decision making problem since each stakeholder has decisions to make before being able to participate in the evaluation process.
- COTS evaluation is carried out when system requirements are still subject to change. This means the evaluation goes on as stakeholders develop deeper understanding of the system under development. In addition, stakeholders assimilate information about COTS alternatives as the evaluation process progresses. This leads to two major issues:
  1. Difference in the rate at which stakeholders acquire relevant information,
  2. Changing stakeholder preferences.
- COTS evaluation benefits tremendously from previously conducted evaluation processes. However, stakeholders can only gain from historical data that were generated by roles similar to their own role(s) in the current evaluation.

The above characteristics require that an appropriate COTS evaluation DSS has, among others, the following capabilities:

1. Distributed problem solving capabilities so as to offer decision support to stakeholders as they develop their views and preferences.
2. Handling iterative decision making both at individual and group level.
3. Evaluation of products without views of some stakeholders and inclusion of those views (views of previously absent stakeholders) when they are available (flexibility and modularity).
4. Distributed learning in order to offer efficient and customized decision support to different stakeholders (customization of the DSS to stakeholders).

Agent technology makes it easy to develop and use a DSS that offers the above capabilities [4]

4.2 An Overview of the Proposed System

We suggest using an agent based system to realize our COTS evaluation conceptual model. Figure 2 is a high level layout of the system with the following components:

1. Methods Knowledge Base (MKB) that provides stakeholders' agents, through the administrator agent, with the knowledge required to use properly different methods and techniques during the evaluation and negotiation processes.
2. COTS Knowledge Base (CKB) that stores information quantitatively about different COTS candidates. This will be discussed in more detail later on in this paper
3. Several agents that represent stakeholders involved in the evaluation process. All stakeholders are assigned agents to ensure getting the best results from the COTS evaluation process. Two roles are assigned to these agents: first, to collect corresponding stakeholder preferences to be used in the evaluation process; and second, to evaluate alternative products according to the collected views of the individual stakeholders and the data obtained from the CKB.

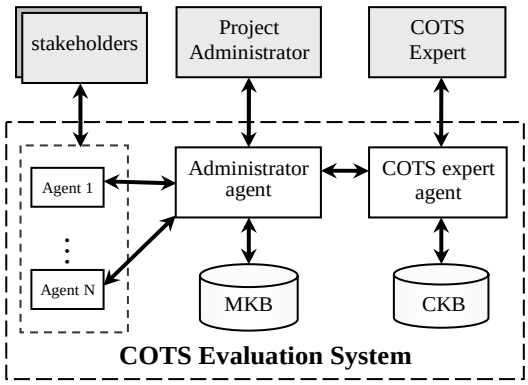

Figure 2. High level structure of our evaluation system

4. An administrator agent, maintained by the project administrator, to manage and coordinate the different tasks throughout the evaluation process (e.g. the negotiation phase).
5. COTS expert agent which has mainly two roles: first, to provide COST experts a fast and easy means to access and manage the CKB. ?should the following sentence go first. We mean by the COTS expert the person who is responsible of managing and maintaining the CKB to keep it up to date with the latest information about COTS products. The second role of the COTS expert agent is to pass on the needed information about COTS products to the stakeholders' agents to perform a sound evaluation process.

4.3 The Benefits of the Proposed System

We believe that by using the proposed system, several benefits will be achieved:

- This system can be used for automatic documentation for different tasks during the evaluation process. Each agent should document its set of goals, plans, tasks, and results. This eases understanding and learning from previous evaluation processes
- When conflict occurs between stakeholders in ranking COTS alternatives, it easy to determine the source and impact of these conflicts before resolving them. This saves time and resources, because negotiation is carried out only if deemed necessary.
- Agent technology allows for easy semi-automation of the negotiation process. Therefore, negotiation results are obtained faster and are mainly dependant upon product characteristics and stakeholder preferences. This contrasts with the *consensual approach* where the negotiation results may depend upon the negotiation skills of stakeholders. This does not mean that the proposed system is intended to remove all negotiations between stakeholders, but rather to give them extra facilities to make faster robust decisions.
- With negotiation automated, it is possible to simulate the COTS selection process, try out different scenarios and weigh their impact on the outcome of the COTS evaluation process.

- The agent-based negotiation process does not require stakeholders to agree upon evaluation criteria or their relative importance. Therefore, it is possible for the decision support tool to accommodate issues related to changing requirements and changing stakeholder needs and preferences.
- Implementing this system as a web-based system allows inheriting many advantages associated with web technologies. For example, stakeholders do not have to be in the same physical place to carry out a sound COTS selection process.
- Having a knowledge base (i.e., CKB) to store and retrieve information during different evaluation processes involves a lot of advantages. For example, it reduces overall required evaluation time and effort. We will discuss that in more detail in the following section.

4.4 Knowledge Bases

Different type of knowledge, models, experiences or lessons learned should be included in LSO knowledge base [11]. In the context of this paper, we split our knowledge base into two, upon which the agents depend to evaluate COTS products. These are the COTS Knowledge Base (CKB) and the Methods Knowledge Base (MKB). In the next two subsections, we will discuss in detail these two knowledge bases.

4.4.1 COTS Knowledge Base (CKB)

In [1], Boehm and Basili stated a hypothesis that almost 99% of all executing computer instructions come from COTS products. Morisio et al [8] suggest that for COTS projects there should be a consulting team whose responsibility is to gather and store information about COTS products. This information can be related to the latest COTS updates, or to the previous selection processes. As a whole, this means that having a repository for COTS products with enough information about them is becoming a necessity. In our proposed approach, we suggest having a CKB as a repository of available COTS products and their characteristics. We believe that having these character-

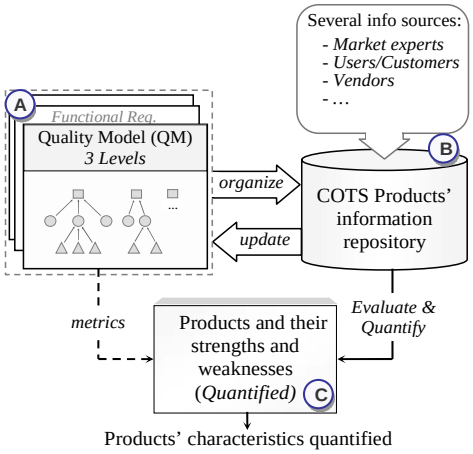

Figure 3. The COTS Knowledge Base (CKB)

istics stored in a quantified form would facilitate and speed up investigating different scenarios, including different aspects and views, for the final selection. However, maintaining and updating such a knowledge base is challenging. Usually, COTS products undergo new releases once or twice a year [1] either by adding new features or modifying existing ones. Hence, we should be very careful about what type of information is to be stored in the CKB. Typically, we should focus on longer lasting features (i.e., those ones which are least probably to change in every release). Nevertheless, all the relevant product features should be up-to-date during the actual development process. Although not the focus of this paper, we intend to address this issue in a future research.

In our proposed system, we use a quantitative information repository of COTS components to help software system developers to more easily and quickly search for and evaluate different COTS alternatives (Figure 3). In general, our approach has multiple layers for sorting and storing information about the products in a very organized way, for example, the *quality features'* layer and the *functional requirements'* layer. In this paper, we only focus upon the quality features of COTS products. However, the same idea can be adapted to store COTS functionality and other features.

The idea of such repository is very simple. In Figure 3, information about each COTS product in the system database is stored in Component B according to a specific pattern provided by Component A. The stored information is evaluated for correctness by an expert(s); then quantified according to some metric defined by Component A; and finally stored in Component C.

Component A: This component represents the pattern with which the information about a COTS product should be stored. It includes different layers for different categories of COTS features. As mentioned, we will focus upon product quality features as an example of how this component is built. The quality model (QM) included in this component is divided into 3 levels: the very high abstract level (L1), the detailed level (L2), and the attribute level (L3) [7]. L1 and L2 represent the high level description of the quality features and are domain independent. They are built according to some global quality standards (e.g., ISO/IEC 9126 [3] and McCall [6] quality models). On the other hand, L3 is the level where certain metrics are provided to measure the attributes affecting the quality features at L2. L3 is domain and application dependent. It represents how L2 characteristics are actually implemented within COTS products. Building L3 should include careful market studies by experts to find out what is available. In addition, Component B continuously updates Component A with the actual attributes available in the market. For example, quality characteristics at L1 and L2 may be "*Reliability*" and "*Recoverability*", while an attribute at L3 may be "*how a COTS handles errors in a given system*".

Component B: this component represents a COTS products' information repository. The information is stored according to certain templates provided by Component A. The CKB is an open system, meaning that it allows different stakeholders (e.g., end-users, customers, vendors, and market experts) to participate and to give their feedback about COTS components. The system has three main interfaces for getting information from different stakeholders:

- An interface for the vendors to submit a detailed description of their products
- An interface available to the public to submit (e.g., through the internet) any experience they have about the product.

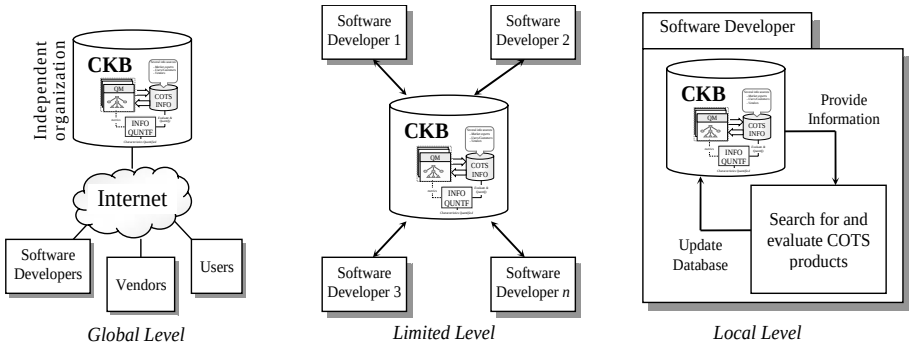

Figure 4: Diffierent levels for using the CKB

– An interface for the software organizations which use the system to have their knowledge, experiences, and test results stored in the system.

The information stored in component B is subjective and is not certified yet. It is stored based upon the subjective view of the people who submitted it.

Component C: this component represents a repository for certified and quantified COTS product features. They are stored according to the patterns and metrics provided by component A. Component C provides quantified information about each COTS product, including its weaknesses and strengths. This allows quick and easy comparisons to be made between different products, especially given that all the data is stored according to the same pattern.

The proposed CKB can be plugged into any COTS evaluation and searching process, such as the system presented in [7] or the system presented in this paper. In fact, it can be used at *one* of three levels (Figure 4):

1. **Local level:** It can be maintained and managed locally by a certain software organization, which stores the history of its COTS selection processes. This will make future COTS acquirement for a specific domain much easier, faster and cheaper.
2. **Limited level:** It can be used as a centralized database between different organizations having an agreement to submit their evaluation results to the CKB.
3. **Global level**: It can be maintained and managed by an independent organization which provides services through a website. This organization should be responsible for testing and ensuring the accuracy of any information regarding product functionality and quality. We assume that financial support for such project is available, for instance from advertisements, from charging vendors who wants to get their product into this KB, or from charging users who search the KB. Of course, the starting budget for such a project will be relatively high, but we believe that with time, this project would be very profitable.

Regardless of level, there should be a search engine capable of searching the database of products and their features according to different criteria and/or keywords. The results should be provided, along with the quality of the product, according to the stored quality model. Moreover, in order to find the best COTS product, the search engine should be provided along with a “compare utility” that performs comparisons between the resultant products from different perspective (e.g., quality, cost, and functionally).

Generally, this knowledge base has the following capabilities:

- Assess the correctness of vendors' claims.
- Provide a quantified evaluation for product functionality and quality.
- Provide information between different COTS products in a comparable way.
- Provide a progressive searching capability for products, meaning that during the searching process, the user can provide further criteria to narrow down the number of results. The search criteria may include:

4.4.2 Methods Knowledge Base (MKB)

This knowledge base, illustrated in Figure 5, has two components. The first component stores the algorithms needed to carry out the evaluation process. For example, in this paper, we rely on bayesian belief networks (BBN) [5] to carry out the evaluation process for individual COTS quality features. The second component is used to facilitate the negotiation process between different stakeholder agents. The negotiation component has three layers; the first layer being the negotiation protocols layer. It includes several rules and methods for the formation of *payoff matrices* and *coalition* [9] between agents. These rules and methods produce input data for the second layer, the negotiation algorithm layer. The second layer includes methods used during the negotiation phase. In the context of this paper, we suggest using the Game Theory [9] to carry out the negotiations. The third layer represents the learning part, using information related to the selection process quality as feedback to improve agents' future capabilities.

- **The "COTS Evaluation Algorithm" component**. This component of the MKB contains information about methods used to evaluate different COTS products. Since we have focused upon quality features in the CKB, we will now focus upon selecting a method suitable for evaluating the quality features in the MKB. For evaluating the quality of different COTS alternatives, we need a method that:
  - Can be used to model and reason about uncertain and incomplete data.
  - Can capture linguistic information about different quality features (e.g. "*the product's ease-of-use*" is high).
  - Can be used to model inter-relationships between problem variables.

  We suggest using BBNs to handle this situation. BBNs are used to model or map real-world situations, which involve uncertain interacting events. They are based upon the Bayesian probability theory. A BBN consists of a direct acyclic graph containing "nodes", which represent different problem variables (events), and are connected by "causal links" to show their dependency relationships. When a node has an effect on another node, the first one is called the "parent node" and the latter one is called the "child node". Each parent node (event) has a conditional probability table (CPT) to show its probability of occurrence according to its child nodes. The CPT is determined by domain experts and historical data.
- **The "Negotiation Algorithm" component**. We suggest using Game theory techniques to handle the negotiations. Products evaluation scores are modeled, according to individual stakeholders' preferences, as payoffs of players (stakeholders), because these scores represent benefits stakeholders get when a particular product is selected. Since agents are in full knowledge of each others utility, we model the negotiation problem as an n-person general-sum cooperative game with complete information.

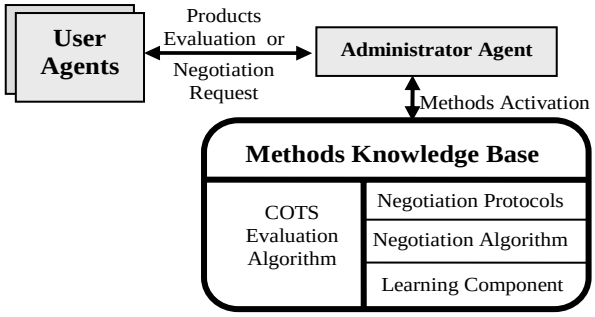

Figure 5: The Methods Knowledge Base (MKB)

If it is determined that the preferences of all stakeholders point at the same COTS product, then there is no need for negotiations between the agents. Ranking of other alternatives is determined by the deviation between their scores and those of the 'best' product. In such a situation, the game is said to have a *core*. The core is defined as set of un-dominated, feasible solution configuration. That is, each element of the core represents a solution configuration acceptable to all players. Otherwise, a game theory solution process is carried out to determine stakeholder Pareto optimal payoffs [9]. The beauty of using game theory negotiation algorithm is the fact that such algorithms are independent of the utility function of stakeholders. Therefore, any multi attribute decision making method can be used for products evaluation algorithm.

5 Conclusions and Future work

This paper reviews a new conceptual model for COTS evaluation. It presents a part of an ongoing research aiming at developing a sound method for COTS selection. We believe that using our approach will give great support to decision makers during COTS selection process. The main focus was on the KB that supports the evaluation process. However, the KB at the global level would be of limited value unless it is possible to establish a community of interested users around it. We also explained how to carry out the other procedures to perform the selection process. The proposed approach has some advantages over similar approaches since: firstly, it uses two knowledge bases, CKB and MKB, to help continuous accumulating, managing and reusing of relevant knowledge. Secondly, it employs agent technology to facilitate negotiations between different stakeholders and providing them with quick alternative scenarios to select from. In addition, we suggest using a hybrid of techniques to address uncertainty and stakeholders varying preferences.

As for the future work, we intend to cover the following points:

- In the proposed approach, we have addressed only the evaluation of the functional and non-functional requirements of COTS candidates. We intend to extend the current approach to cover other relevant issues such as the effect of the amount of effort required for the tailoring and the glue-coding on the final evaluation results.
- We plan to work on integrating suitable models to test COTS candidates at both the product and the system levels.

- This paper is targeted at evaluating a single COTS product. However, in some situations, more than one product is required to cover needed functionality. We intend to broaden our system to address issues of selecting multiple COTS products at the same time.
- We are working on developing guidelines to allow easy management and maintenance for the CKB. Moreover, we intend to define the enabling techniques (e.g. measurements, experiments, etc) [11] for such knowledge base in details

Acknowledgement

The authors would like to thank the Alberta Informatics Circle of Research Excellence (iCORE) and the International Council for Canadian Studies (ICCS) for its financial support of this research.

References

1. Basili, V.,Boehm, B.:COTS-Based Systems Top 10 List,IEEE Computer,Vol.34,No.5.(2001)
2. CeBASE COTS Lessons Learned Repository, available at http://www.cebase.org
3. International Organization for Standardization: Software Product Evaluation - Quality Characteristics and Guidelines for Their Use. Geneve, Switzerland: International Organization for Standardization (1991)
4. Jennings, N.R., Faratin, P., Johnson, M.J., O'Brien, P., Wiegand, M.E.: Using Intelligent Agents to Manage Business Processes. Proceedings of PAAM (1996) 345-360.
5. Jensen, F.V.: An Introduction to Bayesian Networks. Univ. College London Press, London (1996)
6. J. A. McCall. Concepts and definitions of software quality. Factors in Software Quality, NTIS Vol 1, (1977).
7. Mohamed, A., Ruhe, G., Eberlein, A.: Selecting COTS Components for Product Lines. Proceedings of the 4th ASERC Workshop on Quantitative and Soft Computing Based Software Engineering (2004)
8. Morisio, M., Seaman, C.B., Parra, A.T., Basilli, V.R., Kraft, S.E., Condon, S.E.: Investigating and Improving a COTS-Based Software Development Process. ICSE'00. Limmerick, Ireland (2000)
9. Owen, G.: Game Theory. W.B. Saunders Company (1968)
10. Ruhe, G.: Intelligent Support for Selection of COTS Products. Proceedings of the Net.ObjectDays (2002), Erfurt, Springer (2003) 34-45.
11. Ruhe, G.: Learning Software Organisations. In Handbook of Software Engineering and Knowledge Engineering (S.K.Chang.ed.), Vol 1.World Scientific Publishing (2001)663-678.
12. Ruhe, G.: Software Engineering Decision Support - A New Paradigm for Learning Software Organizations. Proceedings of the 4th Workshop on Learning Software Organizations, Chicago, Springer (2003) 104-113.
13. Schwab, D.P., Herbert, G. H.: Assessment of a Consensus-Based Multiple Information Source: Job Evaluation System. Journal of Applied Psychology, Vol.71, No.2 (1986) 354-356.