Global Journal of Computer Science and Technology
Neural & Artificial Intelligence
Volume 12 Issue 10 Version 1.0 Year 2012
Type: Double Blind Peer Reviewed International Research Journal
Publisher: Global Journals Inc. (USA)
Online ISSN: 0975-4172 & Print ISSN: 0975-4350

Unsolved Tricky Issues on COTS Selection and Evaluation

By Zahid Javed, Ahsan Raza Sattar & Muhammad Shakeel Faridi
University of Agriculture, Faisalabad, Pakistan

Abstract - Component Based Software Engineering (CBSE) approach is based on the idea to develop software systems by selecting appropriate components and then to assemble them with a well-defined software architecture. (CBSE) offers developers the twin benefits of reduced software life cycles, shorter development times , saving cost and less effort as compare to build own component. However the success of the component based paradigm depends on the quality of the commercial off-the-shelf (COTS) components purchased and integrated into the existing software systems. It is need of the time to present a quality model that can be used by software programmer to evaluate the quality of software components before integrating them into legacy systems. The evaluation and selection of the COTS components are the most critical process. These evaluation and selection method cannot be resolved by the IT professionals itself. In this study the author tried to compare the twenty three available systematic methods for best evaluation and selection of COTS components.

Keywords : Component Based Software Engineering, Commercial off-the-Shelf, Software Architecture.

GJCST-E Classification : D.2.11

Strictly as per the compliance and regulations of:

Unsolved Tricky Issues on COTS Selection and Evaluation

Zahid Javed [α], Ahsan Raza Sattar [σ] & Muhammad Shakeel Faridi [ρ]

Abstract - Component Based Software Engineering (CBSE) approach is based on the idea to develop software systems by selecting appropriate components and then to assemble them with a well-defined software architecture. (CBSE) offers developers the twin benefits of reduced software life cycles, shorter development times , saving cost and less effort as compare to build own component. However the success of the component based paradigm depends on the quality of the commercial off-the-shelf (COTS) components purchased and integrated into the existing software systems. It is need of the time to present a quality model that can be used by software programmer to evaluate the quality of software components before integrating them into legacy systems. The evaluation and selection of the COTS components are the most critical process. These evaluation and selection method cannot be resolved by the IT professionals itself. In this study the author tried to compare the twenty three available systematic methods for best evaluation and selection of COTS components.

Keywords : Component Based Software Engineering, Commercial off-the-Shelf, Software Architecture.

I. Introduction

Now a day, COTS (commercial off-the-shelf) are widely used in current software development. They are pieces or templates of software that can reused for future concern [1, 2]. COTS can be word processors and email packages etc. [3]. Its selection plays a crucial role in development of final/end product [4]. Selection of COTS means check whether a component is fit or not for a required product [5]. Many challenges and efforts are made for COTS selection process during last decades but no effective solution can be produced or developed which we can say Silver built for it [6]. Different solutions are introduced in different conditions for COTS selection and evaluation.

The objective of this review is as following;

- To evaluate the best technique to find out COTS Selection components.
- To identify currently used decision making practices of COTS Evaluation & Selection.
- Impact of COTS component on developers.
- Problematic COTS integrated with exiting system.

Author α : Dept of Computer Science, University of Agriculture, Faisalabad, Pakistan. E-mail : zahidjaved.uaf@gmail.com
Author σ : Dept of Computer Science, University of Agriculture, Faisalabad, Pakistan. E-mail : uaf_raza@hotmail.com
Author ρ : Dept of Computer Science, University of Agriculture, Faisalabad, Pakistan. E-mail : shakeelfaridi@gmail.com

II. The COTS Selection Process

There is no certified method available for COTS selection [6], some repeated methods are defined. Figure 1 below is showing the General COTS selection process;

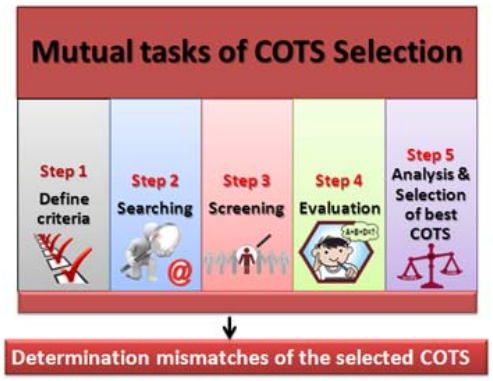

Figure 1: The General COTS selection (GCS) process

Step1: Evaluation criteria should be defined according to stakeholder's requirements.
Step2: COTS product Selection.
Step3: Filtered resultant components based on requirements.
Step4: Short listed COTS then evaluated.
Step5: Analyze evaluated COTS for best fitness. Normally analytic hierarchy process (AHP), used for selection process [7].

After the final step 5 selections of COTS is done to avoid mismatch problem.

III. COTS Selection Approaches

COTS selection and its strategies are compared here. This section shows how different approaches can contribute during selection of COTS.

a) The Evolution of COTS Selection Practices

First proposed by OTSO (Off-The-Shelf Option) approach for COTS selection in 1995 [16]. OTSO was a milestone towards COTS selection where basic structure was defined. Structure was very like to the GCS process described in Figure 1.

In 1996, Kontio published several followup and papers to elaborate OTSO (e.g.) progressive filtering; evaluation criteria includes functionality, non-functional properties, strategic considerations and architecture

compatibility; AHP suggested for comparison.[10,33]

Many approaches were proposed till 1997. These were

i. The IusWare (IUStitiasoft WAR) is addressed selection process and quality evaluation requirements [17].
ii. The PRISM (Portable, Reusable, Integrated, Software Modules) an architecture proposed that can be effective during COTS evaluation [18].
iii. The CISD (COTS-based Integrated Systems Development) model proposed when multiple homogeneous COTS products was required [19].

In 1998 another approach was introduced named PORE. PORE got importance by proposed requirement engineering process for COTS development. It stated that when COTS are evaluated requirement should gather and analyzed [4].

In 1999, several approaches were proposed:

i. The CEP (Comparative Evaluation Process) approach introduced the use of the socalled confidence factor (CF). The more reliable the source of data, the higher a CF value that source gets. Any estimate we make should be adjusted based on the CF value of the source based on which these estimations are made.
ii. The STACE (Social-Technical Approach to COTSE valuation) approach emphasized the importance of non-technical issues, e.g. social, human, and organizational characteristics, during the evaluation process [20].
iii. The CRE (COTS-based RE) approach emphasized the importance of non-functional requirements (NFR) as decisive criteria when comparing COTS alternatives [21].

In 2000, the COTS acquisition process (CAP) which was an evaluation process. This evaluation process (including the evaluation criteria themselves) should be modified based on the effort available. Ochs approach fits the process using expert knowledge [22].

In 2001 a COTS-Aware Requirements Engineering (CARE) approach was introduced [23-26]. CARE used requirements set according to different agents view.

Another set of approaches were introduced in 2002:

i. The PECA (Plan, Establish, Collect, and Analyze) approach from SEI described COTS selection and where to fit that process [27,5].
ii. The BAREMO approach showed how decision can made using AHP method [7].
iii. The storyboard approach advice to apply use case and other visual methods while requirement gathered from customer, and thus get more fitting COTS products [28].
iv. The mutual selection approach aims to select several COTS that evaluated, initial on the narrow level to evaluate each COTS in separation from the others, and then on the overall level to select the finest combination of COTS [29].
v. i-MATE spotlight on middleware selection and the key role is the narrative of reusable requirements for that area [35].

Two more approaches were proposed:

i. The WinWin spiral model risk management approach that can identify, analyze and mitigate risk [3].
ii. Fuzzy logic approach to produce optimal and quantified solutions [30].

In 2004, the Des COTS system, system integrates some tools to classify evaluation criterion using quality models such as ISO/IEC9126 [31,13].

In 2005, the

i. MiHOS (Mismatch-Handling aware COTS Selection) approach was built up [32]. MiHOS relies on the GCS process which handles mismatch issues between requirements and COTS. MiHOS used method like linear programming to make out optimal way out.
ii. Agile COTS Selection method, Some agility concepts were discussed for COTS selection [38].

In 2011

i. CSSP (COTS Software Selection Process) used by organization and software houses [24].
ii. UnHOS COTS-faced uncertainty issues. Uncertain COTS information can be completeness, accuracy, and consistency. Leaving these uncertainty issues will effect COTS quality and stakeholders satisfaction .Figure 2 shows current COTS selection methods tackle the uncertainty dispute and whether these approaches are supported by a tool or not. [39]

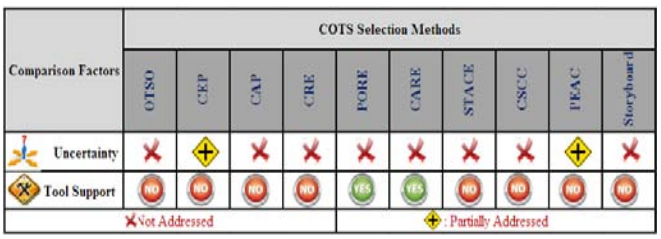

Figure 2 : Comparison of COTS Selection Methods with respect to uncertainty

According to Figure 1, current COTS selections not at all tackle uncertainty. None of these selection methods judge uncertainty in an inclusive style. Only two methods (i.e. PORE and CARE) are hold by a prototype tool. The unavailability of a software tool supporting other COTS selection methods negatively pressures their usability.

There are 23 approaches compare in detail in Figure 3. These approaches in terms of the following criteria:

1) GCS: General COTS Selection method.
2) EVAL: Evaluation strategy used.
3) SNG: Suitability for single COTS selection.
4) MLT: Suitability for multiple COTS selection.
5) MISM: Ability to address COTS mismatches in a systematic way during/after the selection process.
6) TAILOR: Tailor ability of the process based on experts' knowledge. Satisfying this criterion does not necessarily imply the existence of any systematic tailoring techniques.
7) TS: Availability of tool support to facilitate the application of the approach
8) UnHOS: (Uncertainty Handling in COTS Selection) for evaluating COTS candidates while explicitly representing uncertainty. Completeness, accuracy, and consistency

Approach			Comparison Factors							
Sr No.	Name	Year	GCS	EVAL	SNG	MLT	MISM	TAILOR	TS	Uncertainty
1	OTSO	1995	✓	PF	✓	✗	✗	✗	✗	✗
2	IusWare	1997	✓	Any	✓	✗	✗	✗	✗	✗
3	PRISM	1997	✓	PF	✓	✗	✗	✗	✗	✗
4	CISD	1997	✓	PF/PZ	✓	⊕	✗	✗	✗	✗
5	PORE	1998	✓	PF	✓	✗	⊕	✗	✓	✗
6	CEP	1999	✓	PF	✓	✗	✗	✗	✗	⊕
7	STACE	1999	✓	KS/PF	✓	✗	⊕	✗	✗	✗
8	CRE	1999	✓	PF	✓	✗	⊕	✗	✗	✗
9	CAP	2000	✓	PF	✓	✗	✗	✓	✗	✗
10	CARE	2001	✓	PF	✓	✗	⊕	✗	✓	✗
11	RCPEP	2001	✓	PF	✓	✗	✗	✗	✓	✗
12	PECA	2002	✓	Any	✓	✗	⊕	✓	✗	⊕
13	BAREMO	2002	Step5	⊕	✓	✗	✗	✗	✓	✗
14	Storyboard	2002	✗	KS/PF	⊕	✓	⊕	✗	✗	✗
15	CS	2002	✓	⊕	⊕	✓	⊕	✗	✗	✗
16	i-MATE	2002	✓	PF	✓	✗	✗	✗	✗	✗
17	WinWin	2003	✓	PF	✓	⊕	⊕	✗	✗	✗
18	Erol's	2003	✓	⊕	✓	✗	✗	✗	✗	✗
19	DesCOTS	2004	✓	PF	✓	✗	✗	✗	✓	✗
20	Agile	2005	✓	PF	✓	✗	✗	✗	✗	✗
21	MiHOS	2005	✓	KS/PF	✓	✗	✓	✗	✓	✗
22	CSSP	2011	✓	PF	✓	✗	✗	✗	✗	✗
23	UnHOS	2011	✓	PF	✓	✗	✗	✗	✓	✓
PF: Progress Filtering			✓	Full Satisfies the criterion						
KS: Keystone			✗	Does not satisfy the Criterion						
PZ: Puzzle assembly			⊕	Partially, informally, or implicitly satisfies the criterion						

Figure 3 : Comparing COTS selection approaches

IV. COTS Evaluation

It is core for COTS selection because fitness of COTS product based on it. Necessary information is provided in COTS evaluation so select the best fit as per requirement [8, 9].COTS products are evaluated on the base of stakeholders requirements. Suggested hieratically COTS evaluation method in which goals refined according to application requirements and architecture [10]. The practical literature having 6 steps for COTS evaluation according to quality Model [11,12].

Based on the ISO/IEC 9126 quality model [13] the activities defined for COTS evaluation. Three strategies that can applied to evaluate COTS [14, 15]

1. Progressive filtering, Start with large number of COTS and then used iteratively evaluation method, LOW fitted eliminated during each loop. 1to 4 steps used in this strategy in the GCS process repeatedly as desired COTS product is available for system integration.
2. Puzzle assembly, suppose that COTS is like puzzle pieces. This means a COTS product feels bets fit when not integrated but fails to integrate. This shows COTS should be considered in isolation as well as in integration scenario.
3. Keystone identification, starts with requirements identification (e.g. vendor location or type of technology), and searching COTS that fulfills requirements. This enables elimination of not required COTS. One or more strategies are enabled in some projects [15]. Some developer might use 'keystone identification' first and then later 'progressive filtering.

V. Evaluating Existing Approaches

Some issues that are not properly discussed latterly are focused here. Above comparison shows there are varieties for COTS selection. This is open research option:

Problem 1: Best evaluation technique for COTS selection. COTS Selection and evaluation can have following issues.

1. COTS Integration.
2. Mismatch of non-functional requirement for COTS.
3. Indecision Handling (completeness, accuracy, and consistency)
4. Rotating concepts of Stakeholders.
5. Multi Criteria decision-making (MCDM), need for COTS Component.

Different COTS evaluation methods are proposed for different domains. There is no such method available that give solution for above problems. COTS selection is purely base on stakeholders requirements.

Problem 2: Identification of currently COTS selection and evaluation decision making methods. Decision making approaches like Weighted Averages, Fuzzy logic, Bayesian Belief Networks (BBN), Analytic Hierarchy Process (AHP) and linear programming are available. These approaches can apply on Quality selection and evaluation process. IFCOTS software finds then use BBN or AHP, IF COTS component then we used Fuzzy logic or linear programming is used.

Problem 3: COTS components impact on developers.

1. Usually source code of COTS product is not given to developer. This means decision of buy and build

is not easy. Major disadvantage of COTS, it is "black boxes" means not easy to test.

2. COTS component may be mismatched to current system while integrated. Developer should always keep in mind requirements of stakeholder to avoid this situation. Normally COTS product has it specific attributes so it will make misfit while integration to existing system. Two types of mismatches are: Architectural mismatches, and COTS mismatch [36, 37].

Problem 4: Interpretability is another issues occurs when COTS mismatched. This means COTS components mismatches due to lack of adapting quality model while selecting COTS. Using quality model COTS having same standard can match and mismatching can be reduced.

VI. Conclusion

In this paper, we explored the evolution of COTS selection practices, and compared the 23 of the most significant COTS selection approaches. In spite of the great variety of these approaches, there still many open issues related to the COTS selection process that need further research. The objective of this study was to evaluate the best technique to find out COTS Selection components, to identify currently used decision making practices of COTS Evaluation and Selection, impact of COTS component on developers and problematic COTS integrated with exiting system. In future, the author would like to present these models quantitative by using self-completion questionnaire method. This questionnaire method will help out the IT professionals to determine which COTS approach is the best for evaluation and selection of desired components.

References Références Referencias

1. D. G. Firesmith, "Achieving Quality Requirements with Reused Software Components: Challenges to Successful Reuse," inMPEC'05, St. Louis, Missouri, 2005.
2. M. R. Vigder, W. M. Gentleman, & J. Dean, "COTS Software Integration: State of the art," NRC, 39198, Jan 1996.
3. B. Boehm, D. Port, and Y.Yang, "Win Win Spiral Approach to Developing COTSBased Applications, "EDSER-5, Oregon, 2003.
4. N. A. Maiden & C.Ncube, "Acquiring COTS Software Selection Requirements, "IEEE Software, vol.15(2), 1998.pp. 46-56.
5. SEI: Software Engineering Institute in Carnegie Mellon Univ, athttp://www.sei.cmu.edu.
6. G. Ruhe, "Intelligent Support for Selection of COTS Products", LNCS, Springer, vol. 2593 2003. pp. 34-45
7. T. L. Saaty, The analytic hierarchy process. New York: McGraw-Hill, 1990.
8. D. Carney, "Evaluation of COTS Products: Some Thoughts on the Process," SEI Interactive, CMU university, Sept 1998,
9. D. Carney, "COTS Evaluation in the Real World," SEI Interactive, Carnegie Mellon University, Dec 1998,
10. J. Kontio, G. Caldiera, and V. R. Basili, "Defining factors, goals and criteria for reusable component evaluation," in CASCON'96Toronto, Ontario, Canada: IBM Press, 1996.
11. X.Franch &J.P. Carvallo, "Using Quality Models in Software Package Selection, "Software, vol.20(1), Jan/Feb 2003. pp. 34-41
12. J. P. Carvallo, X.Franch, G. Grau, &C. Quer, "COSTUME: a method for building quality models for composite COTS-based software systems," in QSIC'04, Germany, 2004, pp. 214 – 221
13. ISO/IEC1926-1, Software Engineering - Product Quality Model-Part1: Quality Model. Geneve, Switzerland: ISO, 2001.
14. D. Kunda and L. Brooks, "Identifying and Classifying Processes(traditional and soft factors) that Support COTS Component Selection: A Case Study," ECIS'00, Austria, 2000.
15. P. A. Oberndorf, L. Browns word, E. Morris, and C. Sledge," Workshop on COTS Based Systems," SEI Institute, CMU, Special Report CMU/SEI-97-SR-019, Nov. 1997,
16. J. Kontio, "OTSO: A Systematic Process for Reusable Software Component Selection," University of Maryland, Maryland, CSTR- 3478, December 1995,
17. M. Morisio and A. Tsoukis, "IusWare: a methodology for the evaluation and selection of software products," IEE Software Engineering vol. 144 (3), June 1997.
18. R. W. Lichota, R. L. Vesprini, and B. Swanson, "PRISM: Product Examination Process for Component BasedDevelopment," SAST '97, 1997, pp. 61-69.
19. V.Tran&D.Liu,"A Procurement-centric Model for Engineering Component-based Software Systems," SAST '97, 1997,pp.7079.
20. D. Kunda & L.Brooks, "Applying social technical approach for COTS election,"UKAIS'99, Univ of York, McGraw Hill, 1999.
21. L. Chung, B. A. Nixon, E.Yu, and J. Mylopoulos, Non-Functional Requirements in Software Engineering: Kluwer Academic, 99.
22. M.Ochs, D.Pfahl, G. Chrobok- Diening, and B. Nothhelfer-Kolb, "A COTS Acquisition Process: Definition and Application Experience," ESCOMSCOPE' 00, Munich, Germany, 2000.
23. L. Chung and K. Cooper, "A COTS-Aware Requirements Engineering (CARE) Process: Defining System Level Agents, Goals, and Requirements," Department of Computer Science, The University of Texas, Dallas, TR UTDCS-23-01, Dec 2001.

24. L. Chung and K. Cooper, "Knowledge-based COTS-aware requirements engineering approach, "SEKE'02, Ischia, Italy, 2002.
25. L. Chung and K. Cooper, "Defining Goals in a COTS-Aware Requirements Engineering Approach," Systems Engineering journal, vol. 7 (1), 2004, pp. 61-83.
26. L. Chung and K. Cooper, "Matching, Ranking, and Selecting COTS Components: A COTS-Aware Requirements Engineering Approach," in MPEC'04, Scotland, UK, 2004.
27. S. Comella-Dorda, J. C. Dean, E. Morris, and P. Oberndorf, "A Process for COTS Software Product Evaluation," in ICCBSS'02,Orlando, Florida, 2002, pp. 86 - 96.
28. S.Gregor, J.Hutson, & C.Oresky, "Storyboard Process to Assist in Requirements erification and Adaptation to Capabilities Inherent in COTS," in ICCBSS'02, Florida, 2002, pp.132-141.
29. X. Burgues, C. Estay, X. Franch, J. A. Pastor, and C. Quer,"Combined Selection of COTS Components," in ICCBSS'02,Orlando, Florida, 2002, pp. 54-64.
30. I. Erol and W. G. Ferrell-Jr., "A methodology for selection problems with multiple, conflicting objectives and both qualitative and quantitative criteria " International Journal of Production Economics vol. 86 (3), Dec 2003. pp. 187-1999
31. G. Grau, J. P. Carvallo, X. Franch, and C. Quer, "DesCOTS: ASoftware System for Selecting COTS Components," inEUROMICRO'04, Rennes, France, 2004, pp. 118-126.
32. A. Mohamed, G. Ruhe, and A. Eberlein, "Decision Support for Handling Mismatches between COTS Products and System Requirements," in ICCBSS'07, Banff, Canada, 2007.
33. J. Kontio, OTSO: A Systematic Process for Reusable Software Component Selection, Univ. Maryland report CS-TR-3478, UMIACS-TR-95-63, 1995.
34. L. Chung and K. Cooper, "COTS-Aware Requirements Engineering and Software Architecting", Proceedings of the 4th International Workshop on System/Software Architectures (IWSSA), 2004
35. J. Bhuta, B. Boehm, "A Method for Compatible COTS Component Selection", ICCBSS, LNCS vol. 3412, Springer, 2005.
36. P. Avgeriou and N. Guelfi, "Resolving Architectural Mismatches of COTS through Architectural Reconciliation" in the 4th International Conference on COTS-BasedSoftware Systems (ICCBSS'05), Bilbao, Spain, 2005, pp. 248-257.
37. M. Shaw, "Architectural Issues in Software Reuse: It's Not Just the Functionality, It's the Packaging," in ACM SIGSOFT Symposium on Software Reusability, 1995, pp. 3-6.
38. Navarrete, Catalunya "How agile COTS selection methods are (and can be)?" Software Engineering and Advanced Applications, 2005. 31st EUROMICRO Conference on 30 Aug.-3 Sept. 2005 PP- 160 – 167
39. Ibrahim, Abdel-Halim, H. Far, Eberlein3 "UnHOS: A Method for Uncertainty Handling in Commercial Off-The-Shelf (COTS) Selection: International Journal of Energy, Information and Communications Vol. 2, Issue 3, August 2011

This page is intentionally left blank