I.J. Wireless and Microwave Technologies, **2017, 5, 22-31**
Published Online September 2017 in MECS(http://www.mecs-press.net)
DOI: 10.5815/ijwmt.2017.05.03

Available online at **http://www.mecs-press.net/ijwmt**

Selecting the COTS Components Using Ad-hoc Approach

Somya Goyal [a], Anubha Parashar [b]

[a] *Scholar, Computer Science & Engineering, Vaish College of Engineering, Rohtak, India*
[b] *Assistant Professor, Computer Science & Engineering, Manipal University, Jaipur, India*

Received: 16 December 2016; Accepted: 06 August 2017; Published: 08 September 2017

Abstract

This paper presents the current scenario of our software industry which is deploying CBSE approach to construct high quality deliverable software products at shorter time to market. As both Vendor-specific and OSS COTS components are equally popular now-a-days. Hence, the availability of a wide range of COTS components in market is quite high. To select the best suitable candidate among the various available components, various formal methods and techniques like OTSO have been introduced by researchers. In this paper, COTS based software development & SDLC under CBSE tradition are discussed. Along with this discussion, it uncovers the fact that our software developers are applying Ad-hoc techniques as per their taste for making the selection of the most appropriate components for their projects rather than following the formal methods. Through this paper, various possible reasons behind the 'Not-so-In-Use' nature of these formal methods are being reported.

Index Terms: COTS (Commercially-off-the-shelf); CBSE (Component based Software Engineering); OTSO (Off-the-shelf-Option); WSM (Weighted Scoring Method); AHP (Analytical Hierarchical Processing).

1. Introduction

Commercially-off-the-shelf components are being used to develop very high quality products in less time. In this era, "Buy, Don't Build" is being practised. Reuse is the key feature behind Component based Software Engineering (CBSE) to avoid rework and improve the overall quality of the product. COTS framework supports design and construction of software using pre-existing reusable components.

* Corresponding author.
E-mail address:

This paper is organized as follows: Section 2 describes the COTS Technique for software development and its impact on the Software Development Life Cycle (SDLC). Section 3 brings the light on the most sensitive phase of the entire COTS activity framework that is the selection of the most appropriate COTS component among the various offers available in the market. It also gives a timeline of various formal methods introduced to carry out this selection task. Sections 4 unveiling the fact that these formal approaches to make the selection of components are not practical and are not being followed in the current market. Rather, Components are selected in ad-hoc approach as per the requirements or taste of the project. This also highlights the possible reasons behind such trend and practices. Section 5 produces the evidence for the truth uncovered in previous section regarding the selection as per the taste but not using empirical methods. Conclusion and future work are given in the final section.

2. COTS Based Software Development

CBSE is a paradigm that aims at constructing and designing systems using a pre-defined set of software components explicitly created for reuse. It shifts the emphasis from programming to composing software systems. CBSE embodies “the ‘buy, don’t build’ philosophy”. CBD Activities: The main activities in COTS framework are: Search -> Select -> Create/Adapt -> Integrate -> Maintain. Fig 1 illustrates COTS Framework.

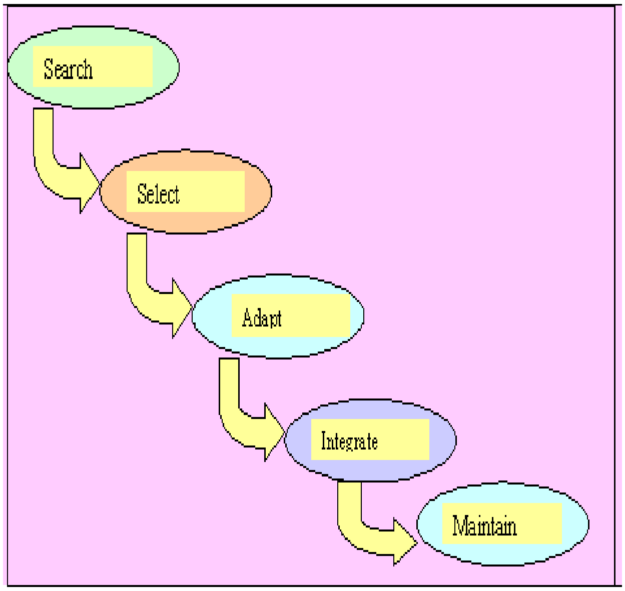

Fig.1. COTS Activity Framework

COTS Components are the parts designed to be included within developed software to provide additional functionalities. These are designed, developed, tested and documented by their vendors. In design, there components are very generic [1]. It follows, “Build once, Reuse often”. COTS has become very common and popular in Software Engineering field because of the various benefits like reuse of pre-fabricated components, fast delivery of the final product to the market and the better quality of the overall software product. According to IDC survey, more than 50 % of Developers follow COTS based approach for software development [2].

The marketplace offers a wide variety of pre-built components which are already tested and documented. The only decision is the appropriate selection of the component from the market as per the requirement specification of the project.

The Fig 2 shows the COTS based development involves range of activities from selection of the COTS component through tailoring & Glue-code development to the final integration to the developmental target Application.

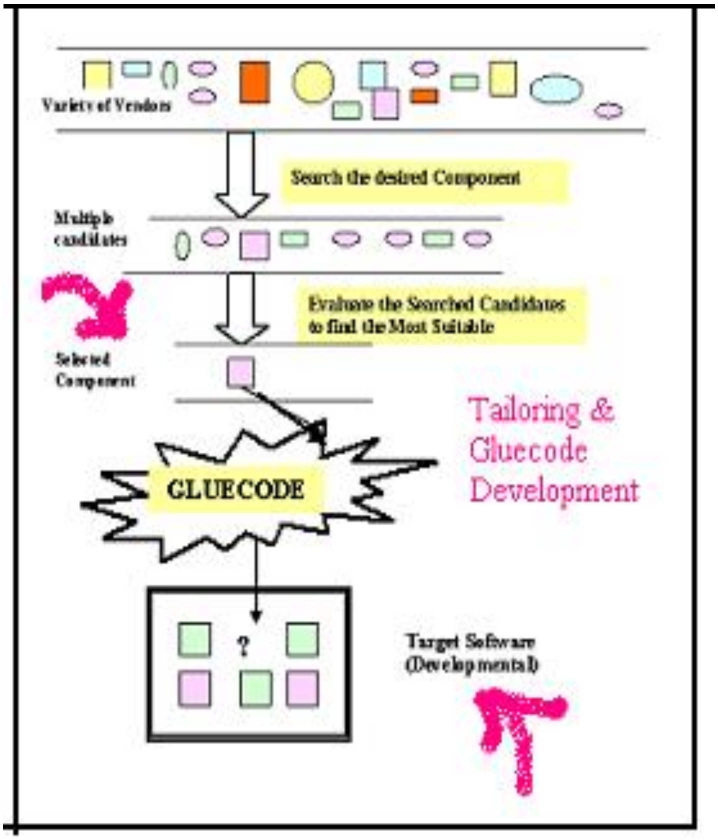
Variety of Vendors
Search the desired Component
Multiple candidates
Evaluate the Searched Candidates to find the Most Suitable
Selected Component
Tailoring & Gluecode Development
GLUECODE
?
Target Software (Developmental)

Fig.2. Glue-code to bind the COTS components

Consider, if the application requires 60% new coding then, Total cost in the development of such software is:

$$E(Total) = E(new) + E(Select) + E(Tailor) + E(Integrate)$$

where

E(new) = Cost of developing new code

E(Select) = Cost in searching, evaluating and selecting the COTS component

E(Tailor) = Cost in tailoring the selected one

E(Integrate)= Cost involved in writing glue-code &

Integrating the component to the system

The Fig 3 illustrates the slight up gradation in the traditional SDLC by COTS activities.

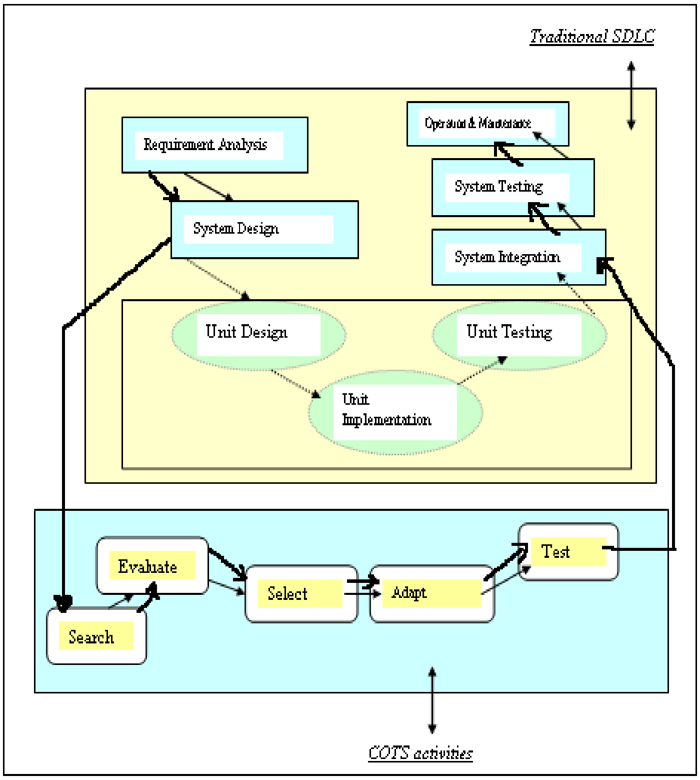

Fig.3. Updated SDLC involving COTS activities

3. Component Selection & Formal Methods

Selection has grown as the most sensitive phase in COTS framework. In the marketplace, vast variety of candidate components is available but, these market offers lack in the precise and detailed documentation of quantitative description of the components. Hence, selection becomes tedious and difficult. Selection of the most appropriate candidate for the target project is becoming sensitive and risk prone. The reason is that a wrong selection may result into an insufficient product or may lead to the ultimate failure for not meeting the user requirements.

Selection process determines the availability of COTS solution that has the potential to provide the needed functionalities and evaluates the solution. Selection involves the search for the desired component in the market, which provides a large number of the candidates. Then, some fundamental screening is performed to discard few components which can not satisfy the needs. The evaluation of components is done to finally find out the most appropriate and fittest component. The Fig 4 depicts that as the evaluation techniques are applied, selection becomes narrower.

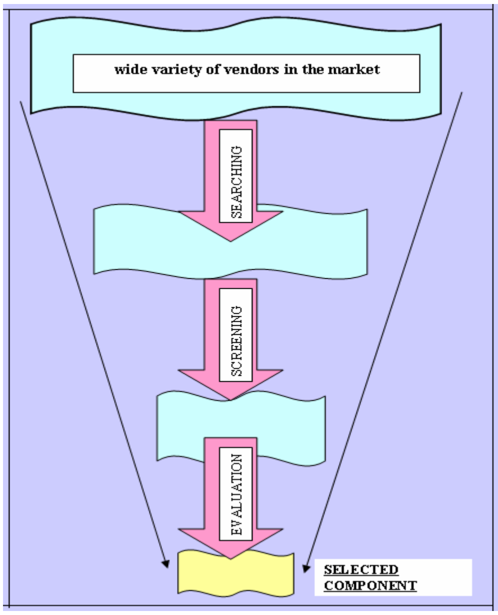

Fig.4. Selection for the COTS Components

Selection involves progressive filtering of the available components to find the single solution which emerges out to be the best after several iterations. This phase is very crucial as a wrong decision may affect the entire project. An immature decision can spoil the entire project and cause the failure. Side-effects of wrong selection may induce errors to the target software, may increase the cost of development rather then decreasing the total effort or may result in an end-product that not serving the customer's desire. The wise and fail-safe decision can improve the quality and efficiency of the product. Along this, it will reduce the cost of development as compared to the by scratch construction. But, the risk is high as if the wrong decision is made, then it will lead to the disaster.

In consideration of the sensitivity and importance of the selection phase, various formal methods were introduced by the researchers summed up in timeline Table 1 shows the timeline of formal methods.

4. Formal to Ad-Hoc Fashion - Reasons

Integrators select OTS Components informally. Various formal methods for 'fail-safe' decision making procedures and methods have been proposed. But, Developers follow Ad-hoc Selection fashion using in-house experience of their experts or web based search engines [4].

The reasons for such practices lie in the fact that developers trust their in-house experts rather than proposed formal methods. It seems somewhat psychological and practical aspect as their experts have better understanding of the concerned project, specific criteria functions and various requirements.

On the other hand, the formal methods are empirical based calculations designed in some generic way. So, this gives the industry a look-and-feel that these methods are not well suited for their specific projects. Another reason is that industry people are so unsure about the pre-conditions for these techniques and the cost effectiveness of such complicated formal methods. Now-a-days, agile development is on the boost. In such scenario, developers may also consider these formal methods posing threat to their specific agile approach of development. Another important reason may be uncovered from the time-to-market pressure over the developers. When there is such intense pressure of running time, they may not take chance of spending so many work-hours in applying these so unsure formal methods just to make out which component is to be selected. Because they consider the selection as merely a single step from the overall task-set to deliver entire product. It seems fruitless to them to spend so much efforts in deploying tedious formal methods just for this only step. It clearly shows that the formal, systematic, empirical and repeatable evaluation methods for the selection of reusable are somewhat not so practical.

Table 1. Timeline of formal approaches for COTS selection

Year	**Method**
1995	**OTSO** (off-the-shelf-option) [3]
1997	**PRISM** (Portable, Reusable, Integrated Software Modules)
1998	**PORE** (Procurement Oriented Requirement Engineering)
1999	**STACE** (Socio-Technical Approach to COTS Evaluation)
2000	**COTS Score**
2002	**CAP** (COTS Adaption Process)
	CRE (COTS based Requirement Engineering)
	CSCC (Controlled Selection of COTS Components)
2003	**CEP** (Comparative Evaluation Process)
2004	**CARE** (COTS Aware Requirement Engineering)
2005	**CCCS** (Compatible COTS Component Selection)
2006	**CSSP** (COTS Software Selection Process)
2008	**Gap A**nalysis [5]
2010	**Ad-hoc** manner [6]
2011	**The State of the Art** [7]
2014	**A-Square** [8]

5. Evidences

Evidences for the non-suitability of formal methods are presented via referring a case study made to select the reusable components in a systematic manner using OTSO [9]. It describes an OTSO experience carried out with Hughes Corporation in EOS program being developed at NASA. This EOS project was looking out for the suitable hypertext browser component for the target system.

OTSO method has 6 stages [3]. OTSO stage Sequence is shown along horizontal axis (named as time) and Number of candidates being considered at a specific stage is shown along vertical axis (named as Number of alternatives).

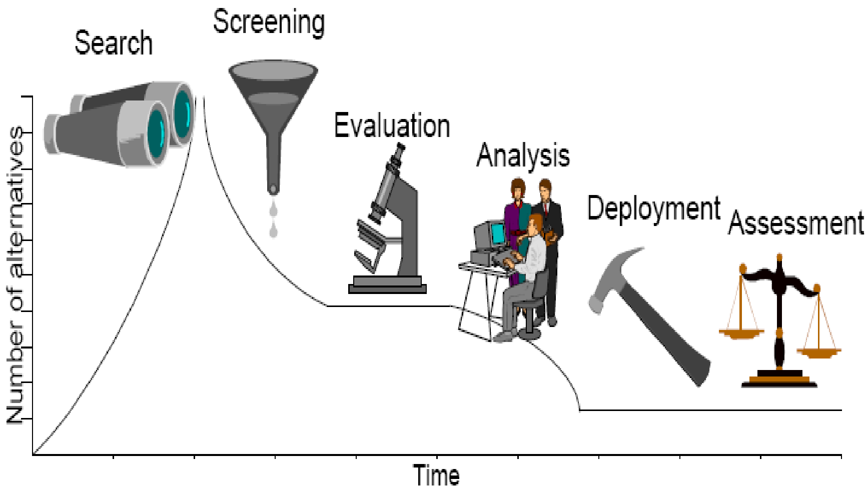

Fig.5. OTSO Stages

5.1. Screening on experts' opinion, in the absence of repeatable formal method in OTSO

Searching is to find out the market availability of the desired component. It is the stage dealing with the largest number of candidates during the entire selection process. Screening is the stage where the count decreases dramatically nearly to the half. Screening is nothing but the 1st filtering on the basis of some basic criteria which are decided by the project specific in-house experts using their experiences. This highlights the very first weakness of the formal approaches that to set the most fundamental & essential criteria for screening can not be stated empirically for any specific project. But, it is achieved with subjective opinion of experts in specific domain in ad-hoc fashion. In the light of case study [10], Searching came with 48 components, then the "key-evaluators" who were amongst the Hughes project personnel set the screening criteria. On the basis of the criteria only 4 candidates were left for further consideration named Mosaic for X, Netscape, Webworks for Mosaic and HotJava. Evidence is so clear that the distance from 48 to 4 was covered on the experts' opinion instead of any formulae. So, it is clear that OTSO is not so capable for defining the project specific screening criteria. Hence, Ad-hoc approach is becoming popular.

5.2. Complexity of WSM and AHP methods

Table 2. WSM results

Criteria/Tests	**Weight Score**	**Weight %**	**Mosaic for X**	**Netscape**	**Webworks**	**HotJava**
Test: Level 2 Compatibility	5	3.4%	3	3	3	3
Test: HTML Level 3 Compatibility	5	3.4%	0	3	0	0
…….	……	……	…..	…..	…..	…..
Test: Required disk space	2	1.4%	1	2	2	5
Test: Ease of installation	3	2.1%	5	5	5	5
Test: Popularity of the tool	4	2.8%	3	5	0	0
Total of Weight Scores	145	1				
SCORE			**470**	**591**	**467**	**427**

Evaluation and Analysis stages involves assigning the scores to the individual candidate, assign criteria weights, perform calculations and make analysis to find the ranking of the candidates. WSM (weighted scoring method) and AHP (analytical hierarchical processing) followed in case study [11] and results are summed up in Table 2 and Fig 6.

It is clear that high complexity is involved in OTSO evaluation & analysis stages either WSM or AHP method followed. This intense complexity and the tedious calculations involved in such methods impart impractical and not-so-popular nature to formal approaches.

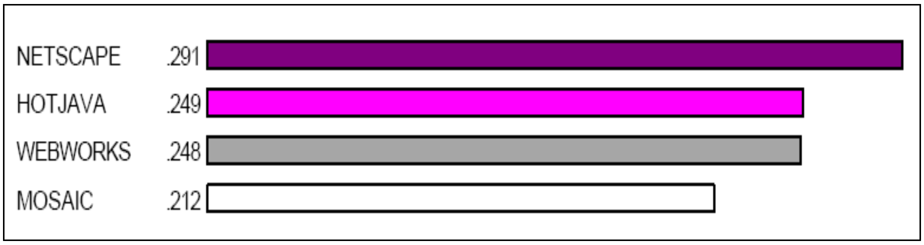

Fig.6. AHP results

5.3. High Effort in selection

Table 3 clearly illustrates that Effort made to evaluate the alternatives and to select the most suitable candidate is quite high.

Such high effort is required in carrying out OTSO activities i.e. 144 hours just to make selection of appropriate option. In Real life Industry, there is short time-to-market pressure on the developers. This scenario demands for the most promising and less expensive method to select COTS components. It leads to exploit the experience of the industry’s most reliable and experienced personnel.

5.4. Unpromising Results leading to High Risk-Factor

Comparison of the results of WSM and AHP shows that the relative rankings among the candidate browser components are different from each other [12]. Both ranked Netscape as the best and appropriate candidate for the project. But the remaining ranking order for tools was different. So, It clearly shows that such high effort consuming methods are not sure shot to select the best. WSM could not distinguish the Webworks from the Mosaic. AHP was sure how to differentiate these two tools. Hence, pre-conditions and cost-effectiveness of formal methods are unsure.

5.5 Current Ad-hoc Selection of COTS Components

Now-a-days, No formal methods is being deployed at the industries, Every Software developer pursuing ad-hoc approach. Kumar et. al. proposed Goal programming approach for making the COTS selection applying upper bound on the cost and lower bound on reliability. They found that suitable to their scenario of Consensus Based Recovery Block Scheme[13]. Gupta et. al. employed Fuzzy interactive approach for optimizing letters [14]. Similarly, Bali et. al . derived fuzzy multi objective optimization model [15]. These are further evidences towards eminent ad-hoc selection of COTS components.

Table 3. Effort estimated in OTSO to select the best hypertext browser.

ACTIVITY	EFFORT (hrs)	%
Search	20	14%
Screening	8	6%
Evaluation	79	55%
Criteria Definition	40	20%
Mosaic for X	10	7%
Netscape	9	6%
Webworks	9.5	7%
HotJava	10.5	7%
Analysis / WSM	5	3%
Analysis AHP	7	5%
Management/administration	20	14%
Learning about methods & Techniques	1	1%
Other (Vendor contact., installation)	4	3%
TOTAL	**144**	

6. Conclusions

COTS based development is current market trend for software exploiting reusability of pre-fabricated components. The selection of suitable component is quite complicated, important and sensitive. Formal methods to select COTS components were launched considering the delicacy of this decision to make it ‘fail-safe’. But, the case study makes it clear that formal methods in pure form are not suitable to our software industry. In-fact, current developers follow ad-hoc methods to select the reusable components for their projects under consideration. As a future work, OTSO like formal methods can be used in ad-hoc manner for selecting components. The comparison among pure formal, pure ad-hoc and formal in ad-hoc approach can also be made.

References

[1] Basili, V.R., Boehm B., “COTS Based System Top 10 List,” IEEE Computer, Vol. 34, No. 5, May 2001.

[2] http:// www.idc.com/getdoc.jsp?ContainerId=IDC_P644, 2007.

[3] J. Kontio, “OTSO: A Systematic process for reusable software component selection,” University Maryland Report CS-TR-3478, UMIACS-TR-95-63-1995.

[4] M. Torchiano et al., “ Overlooked facts on COTS based development,” Software, Vol. 21, No.2, 2004, pp 88-93.

[5] Sheng, J. and Wang, B, “Evaluating COTS Components Using Gap Analysis”, 2008.The 9th International Conference for Young Computer Scientists, IEEE, DOI 10.1109/ICYCS.2008.472.

[6] Couts, C. and Gerdes, P., “Integrating COTS Software : Lessons from a large healthcare organisation”, 2010. IT professional, Vol. 12,pp.50-58.

[7] Tarawneh F., Baharom F., Yahaya J. H. and Ahmed F.,” Evaluation and selection COTS Software process : The State of Art”, 2011. International Journal on new computer architecture and their applications (IJNCAA) Vol. 1, No. 2, pp.344-357.

[8] Mead, N. R. ,: N Evaluation of A-Square for COTS Acquisition”,2014.A technical report, Carnegie Mellon University, SEI. CMU/SEI-2014-TN-003.

[9] J. Kontio, “ A case study in applying a systematic method for COTS selection,” 1996. Proceedings of 18th International Conference on Software Engineering.

[10] J. Kontio, S. Chen, K. Limperos, R. Tesoriero, G. Caldiera and M. S. Dentsch, “ A COTS selection method and experience of its use,” 1995. Proceedings of the 20th Annual Software Engineering Workshop. NASA. GreenBelt, Maryland.

[11] J. Kontio and S. Chen, “Hypertext document viewing tool trade study: summary of evaluation results,” 441-TP-002-001,1995. EOS Project technical paper Hughes Corporation, EOS Project.

[12] J. Kontio, S.Chen, K. Limperos and J. Hung, “Hypertext document viewing tool trade study: Evaluation criteria definitions,” 1995. Internal project documentation. University of Maryland.

[13] Kumar, D., Jha, P.C., Kapur, P.K., and Kumar, U.D., “ Optimal Component Selection Problem for COTS based Software System under Consensus Recovery Block Scheme: A Goal Programming Approach”, 2012. International Journal of Computer Applications, Vol.47, No. 4.

[14] Gupta P., Mehlawat, M.K. Mittal, G. and Verma,S.,” COTS Selection using Fuzzy Interactive Approach: Optimization letters”, 2012. Springer Vol.6, Issue 2, pp. 273-289.

[15] Bali S., Jha P.C., Kumar U.D., “Fuzzy Multi-Objective Build-or-Buy Approach for Component Selection of Fault tolerant system”, International Journal of Artificial Intelligence and Soft computing, Inder-Science, Vol.4, No.2/3, pp. 98-119.

Authors’ Profiles

Somya Goyal is graduated and post graduated majoring in Computer Science & Engineering, from VCE Rohtak. She owns UGC-NET & GATE like certifications and ISOC, IAENG like memberships. Her research interests include Software Engineering, Data Warehousing & Mining, Soft Computing, Computational Intelligence, Machine Learning and Brain Computer Interaction.

Anubha Parashar is presently working as Assistant Professor in Manipal University, Jaipur. She is graduated in Computer Science and Engineering from PDMCE Bahadurgarh and post graduated in Computer Science and Engineering from VCE Rohtak. Her research interests include Machine Learning, Bipedal Locomotion, Humanoid Robotics (locomotion & push recovery), Biometrics Gait, Neural Networks, IOT, Artificial Intelligence and Soft Computing.

How to cite this paper: Somya Goyal, Anubha Parashar," Selecting the COTS Components Using Ad-hoc Approach ", International Journal of Wireless and Microwave Technologies(IJWMT), Vol.7, No.5, pp. 22-31, 2017.DOI: 10.5815/ijwmt.2017.05.03