



Improving COTS-based Software Development Process by Identification and Mitigation of Component Risks

Palak Arora, Amandeep Kaur

Department of Computer Science and Engg.
Lovely Professional University, Phagwara, India

Abstract: - It is very clear that the as the demand of software industry is increasing day by day in every field, the cost of software is also rising very rapidly. In the recent time there is a burden in the IT industry to fulfilling the user expectations with feasible development cost that can provide quality product. So to reduce the complexities of existing system a new strategy developed in software engineering, a component based approach. Component based development (either as COTS components or OSS) aims at developing software by using the existing software components. Time & Cost saving are the main factors to adopt the COTS based approach. This approach improves the software crisis at great level. Because the purpose behind COTS based approach is to develop large systems, by incorporating existing developed components that cut down development time and costs. But the reality of this approach is little different as there are number of problems & risks in components during every development phase. COTS based development approach is still facing many challenges as limited access to internal design due to black box nature of it. The objective of this paper is to provide a theoretical study for the identification of risks during commercial off-the-shelf development life cycle & mitigation guidelines to manage these risks.

Keywords: - Component Based Software Development (CBSD), Commercial Off-the-shelf Software (COTS), Risks

I. INTRODUCTION

As increasing complexities in software industry there was a great need to move in a new direction. Because existing software development process suffers from many difficulties as lesser productivity, high risk, high cost, and maintenance & delivery time, the lack of repeated elements makes software costly, lack of conformity with development standard in industry, make it difficult for applications to interoperate. According to today's requirements the traditional software development methodology is not an ideal deal to meet the current demands of the market. These difficulties create problems in the software industry to build reliable system that provide delivery of products on time within schedule budget. To fulfill the increasing growth of IT technologies and quick delivery demands of clients, Component based software development provides a solution to these problems. Commercial-Off-The-Shelf (COTS) components are best used to meet this need. Due to its attractive features such as cost effective, faster development, efficiency increases, less time to deliver product, budget and schedule saving, more than half of the software developers used COTS components in building their software. The main idea is to reuse already completed components instead of developing everything from the very beginning each time. By adopting CBSD approach, it increases the efficiency & functionality of a system because it promises to provide reliable software in the market. However software engineering industry still faces many problems by adopting CBSD approach. Component- based Development carries many significant risks during the system life cycle. These risks are due to the nature of COTS components, vendor support, technologies and the development process [2].

II. COMPONENT BASED SOFTWARE DEVELOPMENT LIFE CYCLE

COTS- based development (CBSD) is being proposed a low risk development strategy which provides a simple and rapid mechanism for increasing the functionality and capability of a system [1]. It undertakes instant productivity gains, accelerated time to market and lower development cost. COTS- based software development (CBSD) approach is based on the idea to develop new software by selecting the appropriate components from the component repository & then integrating the components into a system as shown in fig. 1 below: -

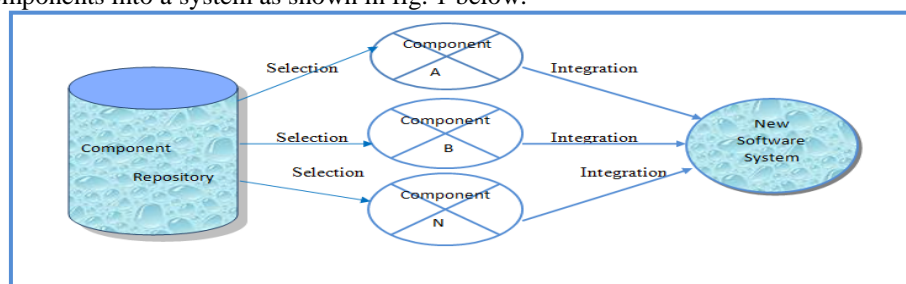


Fig. 1 Component Based software development

The Carnegie Mellon University of Software Engineering Institute (SEI) defines a COTS product as one that is commercially available, leased, licensed, or sold to the general public and which requires no special modification or maintenance over its life cycle [3][4]. In CBSD approach the software systems are built from reusable components in a modular manner, rather than programmed from scratch. The benefits of modularity are that a problem can be decomposed into smaller parts that can be handled independently of each other. So this approach is time-saving approach by using reusable components. CBSD is the process of building the software applications by relying on COTS software components. In general, the COTS-based development (CBD) lifecycle consists of the following phases: identification, evaluation, selection, integration and development of component as explained in fig.2.

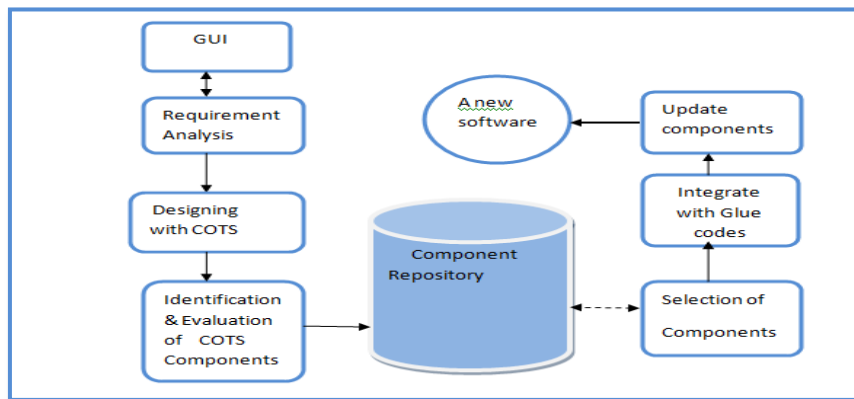


Fig. 2 Phases of COTS- Based Software Development life cycle

The main phases of COTS development life cycle is as: -

1. Analyze the requirements whether these are feasible according to our constraints.
2. Identify & establish Evaluation criteria of COTS components according to requirements.
3. Selection of identified COTS from repository.
4. Integrate different COTS Components with Glue code.
5. Establish or develop new COTS based Software.
6. Maintain & upgrade the COTS based Software/ Architecture.

Benefits & Limitations of using COTS components: - COTS based approach is based on the idea that in this “Component are written only one time can be re-used many a times in software’s”. There are many benefits of using COTS based approach in table2.1:

Table I. Benefits and Limitations of COTS-based development

Sr. no	Benefits	Limitations
1.	Reduced Development Cost	Upfront licensing fees
2.	Improve productivity, quality	Reliability often unknown
3.	Fast delivery of products.	Licensing property procurement delay.
4	Follow technology needs	Multiple vendor upgrades
5.	Devoted support organization	Dependency on vendor
6.	HW/ SW independence.	Incompatibilities among different components.

While utilizing COTS software product provides many benefits such as low cost & instant delivery, there are also many significant risks during the COTS system development life cycle. These risks are due to the lack in system during selection, evaluation and integrating the components.

Risks during COTS- Based Development Life Cycle: -It is very clear that a risk can be any uncertain condition that if it occurs it may have a positive or negative effect on a given objectives, in other words risk is simply the differentiation from the expectation. However, building software using COTS components promises instant delivery to the market and reduction of costs to the software, but still the nature of COTS components are immature. There are certain number of problems and risks using COTS components. These risks ranges from difficulties in establishing the quality of cots software, to predict how the components perform a given context, to difficulties in composing applications, through to problems of managing component-based applications [5]. The risks in COTS components are due to these main factors: -

- a. **The black box nature of COTS software:** - In the COTS based development the main reason of risks is due to black box nature of COTS components. Black box means where Customers do not have access to source code and cannot modify the code to change the properties of the component.

- b. **COTS specifications are usually incomplete:** - Sometimes, documentation is not in a proper way, User manual products documentation does not provide effective description about COTS capabilities and constraints.
- c. **The disparity in the customer-vendor evolution cycles:** - There is no long-term guarantee of support from vendors to the selected COTS components. Sometimes, vendor stops supporting the product or goes out of business.
- d. **Quality of COTS software:** - Sometimes, COTS software does not describe the behavior of the system in response to abnormal input, which is related to quality aspects of reliability.
- e. **The lack of interoperability standards:** - There is a general lack of interoperability standard during the time of integration when same components are implemented using different technologies.

Mainly the risks in COTS development cycle are on these phases: - COTS Identification, COTS Evaluation, COTS Selection, COTS Integration, COTS Development phase. The phases of these risks are as below: -

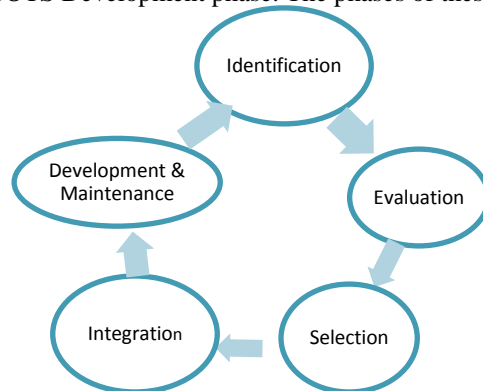


Fig.3 Risks in COTS development life cycle

- A. **COTS Identification phase:** - In COTS development phase, the identification phase start by the process of identify the potential COTS product according to given requirements from repository for future evaluation [6]. Because in the repository there are number of COTS product available, a systematic & well defined approach to identifying COTS is needed to effectively manage the amount and effort to be spent. So after the identification of COTS product prioritization of product is very important as in figure below.

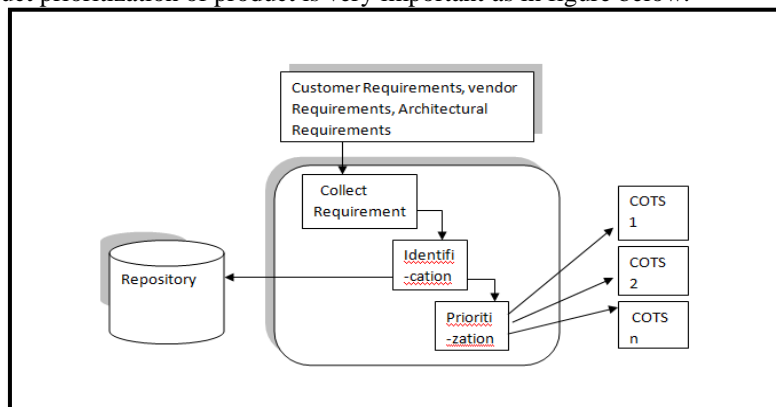


Fig. 4 COTS Identification Phase

a. **Risks during COTS Identification in table 2.2: -**

Table II. Risks during COTS Identification

Risk ID	Parameters	Risks during COTS Identification Phase
RID1	Unclear requirements	Requirements are not understandable to developers.
RID2	Lack of resources	COTS components are not properly identified due to lack of resources.

b. **Reasons for risks in COTS Identification Phase: -**

- ❖ Unorganized COTS identification effort often leads to ineffective use of both resources and time [6].
- ❖ Non-availability or outdated documentation for COTS components.

c. **How to Handle/ Manage these Risks: -**

- ❖ Requirement should be properly documented.

- ❖ Requirements should be feasible within constraints.
- ❖ Latest techniques should be used.
- ❖ There should be a regular watch on repository so that fresh components should be identified.

B. COTS Evaluation and Selection phase: - In the COTS development life cycle when COTS products are properly identified according to requirements, Evaluation and Selection of appropriate COTS product is one of the important steps in COTS components. Component Selection & evaluation is performed in order to select the best fit component. Many developed techniques used for selection & evaluation of COTS components as PORE (Procurement- Oriented Requirements Engineering) method is a template based method to select COTS software, STACE (Social- Technical Approach to COTS Software Evaluation) was developed to address the lack of attention in non-technical issues for COTS software like organization issues and social issues, CRE (COTS Based Requirements Engineering) is an iterative COTS software selection approach that chooses software by rejection [7].

a. Risks in COTS Selection and Evaluation Phase in table 2.3: -

Table III. Risks during COTS Selection and Evaluation

Risk ID	Parameter	Risks during COTS Selection & Evaluation Phase
RS1	Unpredictable	Due to nature of black box, COTS software are difficult to predict.
RS2	Architecture Mismatch	When during COTS Selection Process, Architecture of system is not being properly analyzed.
RS3	Unavailability	Unavailability of the source code for COTS components leads to improper judging of the behavior of the components [2].
RS4	Inflexibility	Rigidity in time constraints of schedule generates inflexibility.
RS5	Unclear Requirements	There is a general lack of Requirements about the system.
RS6	Poor Mapping	Generally requirements of user changes frequently, so the requirement of users and the architecture does not match.
RE7	Lack of methods	During evaluation of COTS products, there is general lack of methods for mapping user requirements to component- based architectures.
RE8	Platform support	In COTS evaluation process, component frameworks offer varying features (eg. Component granularity, interoperability, platform support) [1].
RE9	Market Survey	There is a lack of market survey that can map with user requirements.

b. Reasons to Risks in COTS Selection & Evaluation Phase: -

- ❖ Lack in systematic, repeatable, and well-defined process for evaluating and selecting COTS software [8].
- ❖ Lack of experience to plan for the selection process in detail [9].
- ❖ The evaluation and selection of COTS software is still performed using ad-hoc manners [10], depending on the experiences of developer team or their intuition [7].

c. How to Manage/Handle these Risks: -

- ❖ Follow PORE (Procurement Oriented Requirement Engineering) Method during acquisition of requirements and selection of COTS components that satisfy these requirements [14].
- ❖ Proper System Requirements analysis should be done which include the Eliciting requirements, Analyzing requirements, recording requirements steps.
- ❖ COTS-Aware Requirements Engineering and Software Architecting (CARE/SA) Framework should be used, which supports the iterative matching, Ranking, and selection of COTS components. COTS components are represented as an aggregate of their functional and non-functional requirements and architecture [14].

C. COTS Integration Phase: - It is believed that COTS integration helps to reduce risks and cost in the long-term projects, in which different COTS packages are mixed together by custom develop “Glue Code” [11]. For ex, Office Automation Software, email, messaging system, where the components are bundled as a procedural

library [11]. Many different tools and techniques had been developed to integrating COTS components with Glue Code concept as ACIEP - COTS Integrator Experience with Product, ACIPC - COTS Integrator Personnel Capability. The figure of COTS Integration phase is as below :-

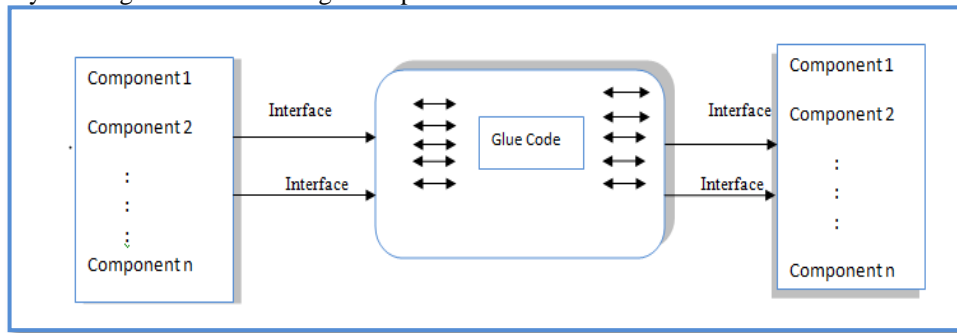


Fig.5 COTS components Integration Phase

a. Risks in COTS Integration Phase in table 2.4: -

Table IV. Risks during Integration Phase

RISK ID	Parameters	Risks during COTS Integration Phase
RINT1	Interoperability	During integrating the components there is a lack of interoperability standards among different COTS Components
RINT 2	Inflexibility	Many times, most component integration process suffers from inflexibility and poor component evaluation scheme. This is a lack of adequate support.
RINT 3	Incomplete	An occurrence of incomplete format among COTS Components
RINT 4	Additional features	Sometimes we need additional feature than required, may exists in selected COTS components which may cause difficulties in integration.
RINT 5	Effort Estimation	Effort for integration may increase from what was estimated
RINT 6	Glue code	Too much glue code critically during Cots integration
RINT 7	Incompatible	When we try to integrate too many incompatible COTS products also cause problems.
RINT 8	Platform Independent	Many Components are not platform independent.
RINT 9	Vendor Support	Vendor may not support over the life time of a product built with COTS components.
RINT 10	Format variations	Components are packaged and delivered in many forms. This cause difficulties and risks during the integration process
RINT 11	Functionality and Performance	COTS components have lack of control over functionality & performance; they may be less effective than they appear.

b. Reasons to Risks during Integration Process: -

- ❖ Lack of requirement specifications.
- ❖ Lack of update & new COTS.
- ❖ Lack in COTS supplier's training and documentation.

c. How to Manage/ Handle these Risks:-

- ❖ Use open Standard technologies that are freely distributed data models or software infrastructure which provide basis for communication and enable consistency among different COTS components.
- ❖ Use of Compositional wrappers can provide a comprehensive treatment of component composition, including improved possibilities for automation of component.
- ❖ Choose exact match of COTS components with system requirements instead of approximate match of COTS components.

D. COTS Development & Maintenance Process: - The objective of development phase is to prepare architecture from the selected COTS components. Many of the problems are faced during this phase because the traditional model is totally different from this CBSD approach [17]. Maintenance risks are due to the improper updating of component repositories.

a. Risks in COTS Development and Maintenance Phase in table 2.5: -

Table V. Risks during COTS Development & Maintenance Phase

Risk ID	Parameters	Risks during COTS Development and Maintenance
RD1	In appropriate process	Risks are associated in development process due to the problems of using an inappropriate development process.
RD2	Lack of code	A new version of COTS software may have lack of code, because sometimes development team is unaware about new techniques.
RD3	Unsuitable	Boehm regards the waterfall model & evolutionary development are suitable for CBD because risk analysis phase is absent in these model [12]
RD4	Insufficiency	Sometimes, resources are insufficient for COTS based development.
RD5	Wrong Estimation	Most of time, the estimation of resources (time & cost) is exceeded during development.
RD6	Lack of Proper Analysis	Sometimes, Risk analysis phase is not considered during selection of COTS components.
RD7	Improper vendor support	Components not properly updated and vendor ceases support.

b. Reasons for Risks in COTS Development phase: -

- ❖ Lack of intelligent development team
- ❖ Unawareness of new technologies and inappropriate selection of Process models.
- ❖ Underestimated Requirements.
- ❖ Lack of Economic resources.
- ❖ The components cannot be upgraded for particular team [12].

c. How to Manage/Handle these Risks: -

- ❖ Proper Resource management is needed [3].
- ❖ Risks driven process model should be adopted.
- ❖ A component repository should be feasible & can help to minimize risks [13].

III. CONCLUSION

At this time, the numbers of applications are built using COTS based approach because it is belief that it is low risk development approach & provide a quality to applications. It is also considered that cost & time saving are main factors to adopt this approach. But in reality COTS based development is facing a lot of challenges & problems. In this paper we present a theoretical study to COTS based development life cycle and highlighted the various risks during COTS development life cycle & explored the main reasons of these risks. Then we provide mitigation advices how to manage these risks. There is still a lack of well defined methodologies & practical tools to improve this approach. So there is a need of appropriate techniques or algorithms to identify and mitigate these risks in an efficient way, so that to provide reliable & error free software to the stakeholders, organizations and in the market.

ACKNOWLEDGEMENT

We would like to thank our Almighty God who has always blessed us, our parents for supporting us in every step of life. We would also like to thank our friends who are big motivator in our life.

REFERENCES

- [1] Awais Rashid and Gerald Kotonya, "Risk Management in component- based Development: A separation of concerns perspective", Computing Department, Lancaster University, UK.
- [2] Rashmi Gupta, Shalini Raghav," Risk Assessment Techniques and Survey Method for COTS Components ", in *International Journal of Software Engineering & Applications (IJSEA)*, vol. 3, No.1, January 2012.
- [3] Amandeep Kaur Johar, Shivani Goel, " Designing of RIMCOTS model for Risk Identification and mitigation for COTS-based Software Development", in *Research Journal of Computer systems Engineering-An International Journal.*, vol02, Issue 02,june,2011
- [4] Szyperski C., "Component Software: beyond object-oriented programming", Addison Wesley Longman; 1998.

- [5] Carina Alves, João Bosco Pinto Filho, Jaelson Castro, “Analysing the Tradeoffs Among Requirements, Architectures and COTS Components”, Centro de Informática, Universidade Federal de Pernambuco.
- [6] Vu Tran, Dar- Biau Liu, “ A Risk Mitigating Model for The Development of Reliable and Maintainable Large-Scale Commercial-Off-The Shelf Integrated Software Systems”, in Proceeding Annual RELIABILITY and Maintainability Symposium, IEEE 1997.
- [7] Feras Tarawneg, Fauziah Baharom, Jamaiah Hj. Yahaya & Faudziah Ahmad, “Evaluation and Selection COTS Software Process: The State of Art”, in International Journal on New Computer Architecture & Their Applications (IJNCAA), ISSN, 2011.
- [8] D.Kunda, “STACE: Social Technical Approach to COTS Software Evaluation”, “Lecture notes in Computer Science,pp.64-84,2003.
- [9] A. Mohamed, G. Ruhe, and A. Eberlein, “Optimized Mismatch Resolution for COTS Selection,” Software Process Improvement and Practice, vol.13, p.157, 2008.
- [10] C. Coutts and P. Gerdes, “ Integrating COTS Software: Lessons from a large Healthcare Organization,” IT Professional, vol.12,pp. 50-58,2010.
- [11] Allen. B. Tucker, “Computer Science Handbook, Second Edition,” available at “books.google.co.in”
- [12] Boehm, B., Abts, C., “COTS Integration: Plug and Pray”, IEEE computer 32(!), Jan. 1999, pp.135-138.
- [13] Jezequel, J. –M. and B. Meyer, Design by Contract: The Lessons of Ariane. IEEE Computer, 1997. 30(1): p. 129-130.
- [14] Lawrence Chung and Kendra Cooper, “Matching, Ranking, and Selecting Components: A COTS-Aware Requirements Engineering and Software Architecting Approach”, available at : <http://www.lsi.upc.edu/events/mpec/articles/chung.pdf>
- [15] L. Jingyue, F. O. Bjornson, R. Conradi and V. B. Kampenes, “An empirical study of variations in COTS-based software development processes in the Norwegian IT.
- [17] Gerald Kotonya and Awais Rashid, “ A Strategy for Managing Risks in Component- based Software Development” , Computing Department, Lancaster University, Uk, IEEE 2001.