International Journal of Information Technology and Knowledge Management
July-December 2011, Volume 4, No. 2, pp. 573-575

COTS COMPONENTS USAGE RISKS IN COMPONENT BASED SOFTWARE DEVELOPMENT

Amandeep Kaur Johar[1] & Shivani Goel[2]

As an increasing number and variety of commercial-off-the-shelf (COTS) become available, it is important to understand the costs, benefits, and risks entailed in using these components. These risks can be due to the nature of COTS components, development process, component technologies and vendor support. There are many issues to be handled while using Component-based software Development (CBSD). The solution is to identify risks during selection, Procurement, Integration and maintenance of COTS components. This paper contributes to identify the various risk factors during component-Based Software development. We examine the risks related to the different development stages in CBSD.

Keywords: Component Based Software Development, COTS, Components, CBD.

1. Introduction

Component-based software development (CBSD) is an emerging development paradigm that promises to accelerate software development and to cut development costs by assembling systems from pre-fabricated components [4]. It promises productivity gains, faster time to market and lower development cost. CBSD also involves significant challenges to constructing software [7].

1.1 Buy vs Build Approach

By adopting a component based development (CBD) approach we have the option of buying off-the-shelf components (COTS) from third parties rather than developing the same functionality in-house. Building a system based on COTS components involves buying a set of pre-existing and proven components, building extensions to satisfy local requirements, and gluing the components together.

1.2 Overview of COTS

The Carnegie Mellon University/Software Engineering Institute defines a COTS product as one that is

- Sold, leased, or licensed to the general public.
- Offered by a vendor trying to profit from it.
- Supported and evolved by the vendor, who retains the intellectual property rights.
- Available in multiple, identical copies.
- Used without modification of the internals[1]

[1,2]ME Student, Asstt. Prof., Computer Science and Engineering Department, Thapar University, Patiala-147004.
E-mail: [1]aman.naaz@gmail.com, [2]shivani@thapar.edu

1.3 Advantages of Using COTS Software

- Shorten Development Time- The application development time will be less because of use of pre-fabricated components.
- Greater ROI- Significant savings can be gained through purchasing COTS components rather than developing the same functionality in-house.
- Enhanced Functionality- In order to use a component containing a given piece of functionality you only need to understand its nature, not its internal details.

So, using COTS components promises faster time-to-market and increased productivity. At the same time, COTS-based software introduces many risks.

2. Risks in Component-Based Development

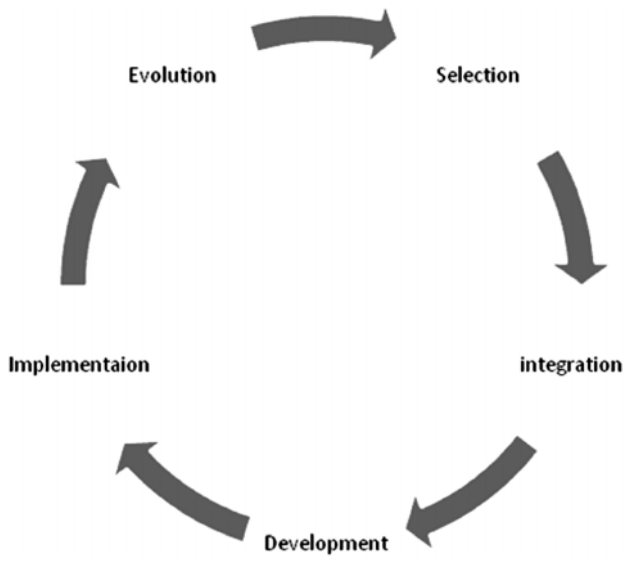
Component-based development poses significant risks to organisations intending to adopt the technology.

These risks come from following factors:

- The black-box nature of COTS software(source code unavailable)
- The quality of COTS software.
- The lack of component interoperability standards.
- The disparity in the customer-vendor evolution cycles [5].

Component developers, application assemblers, and customers must know CBD advantages and disadvantages before developing component-based applications.

The risks in using COTS are categorised on the basis of five application development activities: component Selection, Component integration, development process, and system implementation and system evolution.



Risks in Various Application Development Stages

2.1 Risk Identification in Component-based Development

We will identify various component risks that can occur during various application Development phases. We divide the risks into five main categories and then identify various risks for each individual risk category. Selection risks are associated with problems of evaluating off-the-shelf components for use in system development. Integration risks relate to the problems of composing systems from COTS components. Development risks are associated with the problems of using an inappropriate development process. Implementation risks may stem from the perceived quality of COTS components and various security issues.Evolution risks are related to the extended development and management of component-based applications.

2.1.1 Risks During COTS Selection

COTS selection activity involves COTS identification and COTS Evaluation. Various Risks during this selection phase can be:

- Difficulty in predicting component behaviour due to black-box nature of COTS.
- Unclear system Requirements.
- Lack of cooperation from users.
- Poor mapping of user requirements to component-bases architecture.

2.1.2 Risks During COTS Integration

component integration process may suffers from inflexibility and poor component evaluation schemes. Various risks under integration phase can be:

- Lack of interoperability standards between various components.
- COTS software is generally not tailorable or "plug and play" and hence wrapper code must be generated.
- Effort for Integration increases.
- Cost for integration increases as it requires glue code.

2.1.3 Risks During Development Phase

These risks may be arises due to Development process used in CBD i.e. software development process used might not be suitable for CBD. Boehm regards both the waterfall model and evolutionary development unsuitable for CBD because risk analysis phase is absent in these models [3]. So, we should use process models which include risk analysis phase.

2.1.4 Risks During System Implementation

Risks during this stage can be further categorized into two categories: quality risks and security Risks.

Quality Risks

- unknown design assumptions of COTS components reduces quality of testing.
- The perception of quality may vary across COTS software vendors and application domains.
- may affects system safety or security.
- effects on system performance.
- reduces system reliability.

Security Risks

- System can be used in unintended way [2]. So, sometimes use of COTS software introduces a vulnerability risk.
- Increased external exposure (By connecting the system to the Internet, exposure expands to a large number of potential external).
- Increasing vulnerability and attack options by integrating components with one another.

2.1.5 Risks During System Evolution

- Problems in system updation.
- Problems in licensing arrangements of COTS components.
- Difficulty in COTS replacements.
- Incompatibility of new version with user requirements.

- Vendor goes out of business.
- New version of COTS may provide additional undocumented capabilities.
- Loss of support and continuity when the manufacturer or a product line with "guaranteed" availability cease to exist [6].

3. CONCLUSION AND FUTURE WORK

In this paper we have presented a risk identification approach for component-based development. We have discussed a number of risks in various component-based development stages and these risks arise due to the widespread belief that it is a low risk development strategy. We have concluded that in fact there are a number of risks associated with component-based development. We have identified the various types of risks. This identification has made the risks more visible at each development stage making it possible to carry out activities that can minimise their effects. We acknowledge that a lot more needs to be done in the area of CBD. Future work involves establishing a set of mitigation strategies for the risks identified during various component-based development activities to take advantage of COTS technology.

REFERENCES

[1] John R. Blanchette Sr., "Pros And Cons Of Using Cots Products", IEEE, 2005.

[2] Ulf Lindqvist and Erland Jonsson, "A Map of Security Risks Associated with Using COTS", IEEE Computer, 31, No. 6, June 1998, pp. 60-66.

[3] Boehm, B., Abts, C., "COTS Integration: Plug and Pray", IEEE Computer, 32(1), Jan. 1999, pp. 135-138.

[4] Szyperski C., "Component Software: beyond Object-oriented Programming", Addison Wesley Longman; 1998.

[5] Gerald Kotonya and Awais Rashid, "A Strategy for Managing Risk in Component-based Software Development", IEEE, 2001, pp.12-21.

[6] Susanne Jantsch, "Risks by Using COTS Products and Commercial ICT Services",March 2000.

[7] Padmal Vitharana, "Risks and Challenges of Component-Based Software Development",Commun. ACM 46, 8 (Aug. 2003), pp.67–72.