Selecting COTS Anti-Virus Software for an International Bank: Some Lessons Learned!

Cornelius Ncube & Neil Maiden
Centre for HCI Design, City University London, EC1V 0HB, UK
C.Ncube@soi.city.ac.uk

Abstract

Successful selection of COTS software is difficult and requires tight integration of 2 processes - multi-criteria decision making and requirements engineering- which are both complex and problematic. The selection of COTS software is a multi-criteria decision-making process. However, it is problematic because the selection criteria (i.e., the requirements) are often incomplete and interdependent, and one COTS software rarely satisfies all of the criteria. However, when selecting anti-virus systems, there are added problems. The reason is that for the anti-virus systems, is that you can not just run and test the software as you would do in other systems. This paper reports some lessons learned in the selection of an anti-virus software for a merchant bank. These lessons were later incorporated into a methods for COTS software selection.

1.0 Introduction

This paper reports results of a case study to select a COTS software package for an international merchant bank who were installing an anti-virus security system at its international head office in London and at branches throughout the world. A detailed description of this case study are reported in Ncube & Maiden 2000. The rest of the paper is organised as follows. Section 2 gives a brief overview of the method that was used in this case study; section 3 describes brief background of the organization and detailed description of the case study. The paper concludes with some lessons learned.

2.0 The PORE Method

The PORE method was developed to cover a significant set of requirements acquisition and product selection issues for COTS-based systems development. The PORE (Procurement-Oriented Requirements Engineering) method integrates techniques for requirements acquisition and product selection with process guidance for choosing and using each technique. The method drew on techniques from different disciplines including:

(i) knowledge engineering techniques such as card sorting and laddering (e.g., Rugg & McGeorge 1995) which are useful when acquiring information about categories of products, suppliers, procurement contracts and hierarchical information about product properties as well as the requirements themselves;

(ii) techniques from feature analysis (Kitchenham & Jones 1997) to aid when scoring the compliance of each product to each requirement;

(iii) MCDM techniques (e.g., Saaty 1990) and outranking methods to aid decision-making during the complex product ranking and selection process;

(iv) design rationale techniques (e.g., Buckingham-Shum & Hammond 1994) to record and aid this decision-making process.

The method also included guidelines for designing product evaluation test cases and organising effective evaluation sessions. It supports iterative requirements acquisition and product selection/rejection until one or more products are compliant with a sufficient number of customer requirements. It divides this process into stages and provides three templates for three key stages of the process. Each template defines the product information and customer requirements to acquire, and the techniques for acquiring this information and making decisions about it. At the beginning of the process there are few customer requirements but a large number of candidate products. Guided by the templates, non-compliant products are filtered out using different techniques that are provided in the PORE method box. Over some time, and after a number of iterations, the number of customer requirements increases and the number of candidate products decreases as products are rejected. The 3 templates are:
• Template-1, to guide the requirements engineer when acquiring essential customer requirements and product information sufficient to select and reject products as a result of supplier-given information;

• Template-2, to guide the requirements engineer when acquiring customer requirements and product information sufficient to select and reject products from supplier-led demonstrations using test-cases for individual requirements;

• Template-3, to guide the requirements engineer to acquire customer requirements and product information sufficient to select and reject products as a result of customer-led product exploration, (i.e. user trial).

The iterative nature of the requirements acquisition and product selection processes means that each template might be used several times during a product selection process. The following section describes the use of Template 1 and Template 2 in a real-world banking domain. The part of the template applied is indicated by T1.x or T2.x (e.g. T1.1 or T2.1)

3.0 Organisation Background

The organisation was an international merchant bank installing an anti-virus security system at its international head office in London and at its branches throughout the world. The bank was experiencing a number of security problems throughout its operations. The bank was therefore looking to increase its security by implementing a robust anti-virus system that will meet all its security requirements and needed advice about how to proceed with purchasing an off-the-shelf system.

An evaluation team was set up to find and recommend a suitable system to the bank management. The team used the method templates in the selection of the preferred system. Template 1 guidance was used to identify candidate products by conducting a market survey (T1.1). Techniques that were applied included:

(i) browsing the internet and sending a general call on it to special interests groups who provided a list of suppliers and users of anti-virus software;

(ii) contacting and visiting customer organisation both within and outside the banking sector who have implemented similar anti-virus systems;

(iii) visiting trade shows and exhibitions. A large number of anti-virus suppliers were identified at the shows and a large quantity of supplier and product information was gathered;

(iv) reading relevant computing journals. This helped to identify security products that were being advertised and published independent surveys about the capabilities of the current anti-virus products;

(v) requesting marketing information and product demonstration copies directly from the suppliers.

In parallel to this, the team also identified and acquired core, essential customer requirements (T1.2). Two techniques recommended in template 1 were used:

(i) brainstorming – a brainstorming session was conducted with all stakeholders at the bank’s London branches. The brainstorming session lasted for about three hours and at the end, a list of essential requirements were identified;

(ii) interviews – after the brainstorming sessions, a number of structured and unstructured interview sessions were held with senior IT officers of the organisation’s international division who were responsible for procuring the organisation’s software products. Interviews were also held with the IT manager of the London office to determine their local requirements. On average, each interview session lasted for about two hours.

During the product identification phase, a total of 49 products available on the market were identified. The first-pass essential customer requirements acquired during the brainstorming and interview sessions were structured into a questionnaire that was sent either by e-mail, fax or post to all 49 candidate suppliers (T1.3). The questionnaire asked suppliers to indicate the degree of compliance of their products to each of the essential customer requirements. The questionnaire was divided into 5 sections:

(i) basic product information,

(ii) supplier information,

(iii) product’s requirements coverage,

(iv) supplier’s technical support arrangements, and;

(v) contract conditions.

A covering letter was also sent with the questionnaire. The letter stated the deadline for responses to the questionnaire and explained the process to be used for initial evaluation, and
deadlines for being informed of the evaluation team’s decision.

Of the 49 questionnaires sent out, only 5 suppliers responded by the deadline time. Using the information provided by the suppliers in their questionnaire responses, the team evaluated the five products against atomic customer requirements (T1.5). To achieve this, the main stakeholders were asked to prioritise each requirement as essential, desirable or optional. After this paper evaluation, 1 product was rejected and eliminated from the candidate list as it did not comply with most essential customer requirements (T1.9). The four remaining products progressed to second stage of the evaluation – the supplier-led product demonstrations (T2).

To organise and conduct supplier-led product demonstrations, the team used guidance from PORE’s template 2. The four suppliers were invited, to demonstrate their products to the evaluation team. Before each demonstration session, the team developed acceptance test cases using atomic customer functional requirements, (T2.1). During each product demonstration, the presence of a product feature and that feature’s degree of compliance with atomic customer requirements was sought using test cases (T2.5). The team members awarded scores between 0 (not present or non-compliant with customer requirement) and 7 (present and fully compliant with customer requirement) to each required product feature (T2.7). All the demonstration sessions were tape and video recorded (T2.9). Template 2 also recommends that a stakeholder representative be present during all the demonstration sessions (T2.2). However, this was difficult because relevant stakeholders were not always available.

After each product demonstration session, the individual team member’s scores were collated and an overall score of the product was produced. To determine the best-fit product, the main stakeholders were asked to rate each requirement in percentages to indicate their importance. After the product demonstrations, the product-requirement compliance scores were then multiplied by the percentage rate score to determine the overall ranking of each product. After producing overall ranking of the products, one product was strongly recommended for trial use within the organisation. The product met most of the organisation’s requirements although the other two products had high scores and were recommended as possible alternative solutions. Figure 1 shows the activities that were performed and the templates that were applied during these activities.

4.0 Study Results

The exercise succeeded in recommending a software product to the bank. The method templates were applied to achieve this success. The templates were very useful in acquiring customer requirements, identifying candidate product and suppliers, gathering supplier and product information, and evaluating and selecting products. The method also led stakeholders to be more involved in the COTS software selection process and was effective in systematically filtering out COTS products that were non-compliant with customer requirements.

However, stakeholders also identified some limitations of the method:

(i) demonstrations sessions were sometimes too long and too much time elapsed between the start and end of the COTS software selection: one stakeholder expressed a desire to complete all demonstration sessions in the same week;
(ii) supplier questioning by the evaluation team was too aggressive and intimidating;
(iii) some test cases were sometimes repetitive. For example, compliance with one requirement during product demonstration was tested several times, and this put excessive pressure on the software demonstrator. Stakeholders believed that this slowed evaluation and felt that this could have been tested at a later stage using evaluation copies of the software and the focus during the evaluation sessions should have been on core functional requirements.

4.1 Some Lessons learned

However, the use of the first version of PORE was not completely successful. For example there was a very low response to the questionnaire by the suppliers. PORE does not provide any guidance on what to do if this happens. Other lessons learned about the PORE method are:
Lesson-1: use the questionnaire in combination with other techniques to elicit initial supplier and product information;

Lesson-2: the questionnaire must be short and precise. One of the problems that might have affected the supplier responses is that the questionnaire was too long (15 pages). Suppliers may not have been willing to invest a lot of effort and time without being certain that there will be some benefit to them in the end;

Lesson-3: ask product documentation to be included with the questionnaire response. This will enable the team to compare the results provided on the questionnaire response with actual product descriptions;

Lesson-4: determine the scope and boundary of the product being evaluated. It was very difficult to determine or understand how other systems were associated with the products being evaluated. This made developing test cases very difficult due to the fact that the requirements engineers were required to have knowledge of these systems;

Lesson-5: there is often little time available for COTS software selection. In contrast, the application of the templates can take longer than the available time. The method needs different versions to enable it to be applied more quickly in different circumstances.

4.1 Final Assessment

There is one general lesson learned from this case study:

Lesson-6: be aware that not all requirements can be accurately tested due to legal issues. The anti-virus association prohibits anti-virus vendors to deliberately introduce virus to customer machines for the purposes of testing. This made it difficult to test the effectiveness of the products’ virus detection capabilities.

5.0 Conclusion

This paper reports key lessons learned from evaluations and selection of anti-virus COTS software systems. One lesson is clear: requirements are important for selecting the right COTS software BUT be aware, not all requirements can be easily tested due to legal reasons! One of the main lesson that came out of this case study was that you can not just run and test anti-virus software like any other COTS software as recommended by most evaluation and selection methods. There are legal implications that need to be taken into account. During this case study, virus detection capabilities of the evaluated software packages could not be tested for legal reasons. The anti-virus association prohibits intentional affecting software product with known virus for testing purposes.

Our suggested solution was to involve the anti-virus association as part of the evaluation and selection process. There are a number of ways how this can happen with the 2 most favoured being:

(a) have an anti-virus association member as part of the selection team,
(b) anti-virus association have controlled test suits for testing their member’s software system. The selection process can be extend so as to incorporate an activity where by the anti-virus software’s detection capabilities are actually tested at the association’s site.

These limitations, and the lessons learned were taken into account in the improvement and future designs of the SCARLET method (Maiden and ) and REACT (French and Maiden 2003). The SCARLET method is extensively reported in http://www.atc.gr/banksec/

6.0 References


