Supporting the allocation of software development work in
distributed teams with multi-criteria decision analysis

Ana Barcus\textsuperscript{a}, Gilberto Montibeller\textsuperscript{b,}\textsuperscript{*}

\textsuperscript{a}TD Securities, Toronto, Ontario, Canada
\textsuperscript{b}Operational Research Group, Department of Management, London School of Economics, England, UK

Received 16 October 2004; accepted 26 April 2006
Available online 20 November 2006

Abstract

In the fast moving software industry, projects have been increasingly developed by distributed teams, which are located in geographically remote offices and collaborate using information communication technologies. In such environments, project distribution presents specific challenges, as work in distributed teams increases project technical complexity, communication lines multiply and stakeholders’ interests may be divergent. Despite the importance and complexity of this type of problem, it seems that there is a lack of reports, in the literature, of systems that could support these decisions. This paper presents a real-world case study, where we developed a multi-criteria model for supporting the distributed team work allocation decision for a major global software company. It was developed with a group of software development project managers, using decision conferencing and multi-attribute value analysis. The model deals not only with software engineering attributes, but also “soft” and strategic issues, like team satisfaction and training opportunities. We also discuss some issues and challenges faced during this modelling process.

\textsuperscript{*}Corresponding author.

E-mail address: g.montibeller@lse.ac.uk (G. Montibeller).

Keywords: Software development; Project allocation; Multi-criteria decision analysis; MAVT; Decision conferencing

1. Introduction

Development and global spread of information communication technologies opened up possibilities for collaboration between software development teams in geographically remote offices. This gave rise to the formation of distributed teams and created opportunities for restructuring traditional working ways to benefit from nations’ different competitive advantages.

More and more companies in the UK are turning to the use of distributed teams in the development of software projects aiming to reduce costs and increase development speed \cite{1}. However, this strategy creates new challenges for software engineering, one of them being project and work allocation between remote teams.

This paper describes a real-world intervention where we employed a multi-criteria decision analysis (MCDA) method, in a decision conferencing approach, to address project distribution in a distributed team for a global software development organisation. Working with a group of software engineers and project managers, we applied multi-attribute value analysis to build...
a decision model that supports work allocation in software development projects.

While MCDA has been employed for project allocation in more traditional fields of engineering (e.g., [2]) recently there are suggestions that it may also be useful for supporting problems related to software engineering. For example, Lai et al. [3] suggest the use of an MCDA model to support a group of software engineers in selecting software applications; Wang and Lin [4] propose a method to define which items should be included in software to be developed; Ruhe et al. [5] propose the use of the analytical hierarchy process (AHP) to prioritise software requirements; while Stamelos and Tsoukiás [6] employ a multi-criteria model to evaluate the quality of software. However, as far as we know, there is no previous attempt of using MCDA in supporting software development work allocation in distributed teams.

On the other hand, while the literature on software development is very extensive, there is a lack of attention to the problem of allocating projects in distributed teams. Indeed, research in applying decision making methods in the area of software development is still not abundant, despite a growing interest on software engineering decision support systems (see [7] for details). MCDA techniques have been successfully applied to assist in selecting software development projects from the investment point of view (e.g., [8]) but again they do not address the problem of project distribution in a distributed team.

We hope that this paper may be of interest to both MCDA researchers and practitioners for two main reasons. Firstly, as it describes in detail the process of building the model with a group of decision-makers, it may be a source for further reflection on the practice of MCDA in real-world interventions, a subject that has received growing attention by the MCDA community. Secondly, as the paper deals with a problem that, so far, seems to be ignored by the literature on software engineering—and which has been more and more experienced by many software companies—it may show a possible way of supporting decisions for similar settings.

The paper will start with a presentation of the decision setting and a discussion of the main features of the unsupported decision-making process (which the company was conducting before our involvement with them). These two initial sections set the scene for our intervention. The subsequent section describes in detail both the modelling process and the MCDA model that was developed. Finally, we conclude the paper discussing the outcomes of the intervention, drawing some reflections about the modelling process and suggesting directions for further research.

2. The decision setting

This section presents the context for the decision situation that led to our intervention. It briefly describes the company, its software products, clients and global presence. The organisational structure and project management methodology play a significant part in understanding how project work can be allocated in this distributed team and hence influence the decision support model development that will be described later.

2.1. The client company

The client company is a well established global software house that develops, implements and supports a variety of software products. It is a market leader in providing a comprehensive range of investment- and workflow-management software solutions and related services. The company has been in operation for over two decades, growing organically and through acquisitions, increasing its presence across continents and becoming a global organisation, with development centres in three strategic locations (UK, Australia and a country in Asia) and with direct sales and support offices in all major financial centres in the world. The company is part of a large group of companies, with the total revenue in 2002 of 2.5 billion dollars. (To respect the confidentiality agreement, some details about the company were omitted or modified.)

The suite of software products that the company offers consists of a number of independent software applications, here referred to as “modules”, which interface with each other. Each module addresses a different business area and seamlessly integrates in a product suite, transparent to the final user. The company’s external clients are major financial institutions that often use multiple modules.

2.2. The decision problem

The suite of software products in this company is upgraded bi-annually. An upgrade release consists of a number of interdependent projects that are developed separately through a full project life cycle and then integrated and delivered together.

Each module has a dedicated development team that consists of business and technical experts required to complete a full project life cycle. Development teams report to their project managers, and project managers
report to the development centre manager. In this setting, there is very little or no cooperation between the module teams as modules are developed in relative isolation, and integrated in a separate system integration activity following the completion of module testing.

To respond to high demand for software development and increase capacity while reducing cost, the company decided to use its Asian Development Centre (ADC) to take the “overflow” work from the UK office. The strategy was implemented through the creation of a distributed team between these two centres and the development of some projects in a distributed team instead of module centred teams. The two development centres differ in their maturity, skills sets, size and both corporate and national culture. A new position was created to assist in the development and functioning of the distributed team, here referred to as a Transition Manager.

Taking the overflow work from UK Development Centre (UKDC) means that projects considered for transfer to ADC are enhancements/modifications to the modules that are currently “owned” by the UKDC. These projects can be divided between the two offices in the following manner: a project can be completed in its entirety in one office; a project can be divided by activities where one or more activity would be completed in one office while the others are completed in the other office; discrete tasks could be distributed between the offices. In either case, a decision needs to be made about which project/activity or task will be developed in which office.

2.3. Complexities of this decision problem

The projects that the company develops are all unique and vary in their business requirements, technical complexity, size and other software development project properties. They have an inherent complexity as they are additions to the existing system and often have many interface points.

Transferring projects from UK to the Asian development centre increases the structural complexity [9] of the project by increasing the number of its elements and the interdependences of its activities. The project environment area becomes more uncertain with managing office utilisation and development of two software development teams in parallel and as a distributed team. The goals of the project also become more diffuse, as there is an increased number of stakeholders and the project allocation impacts on general strategies of the company (like the objective of developing the ADC). Increased project complexity resulted in increased uncertainty to project success and generated problems in the unsupported decision-making process, as we discuss in the next section.

3. The unsupported decision-making process

In this section, we describe how the company was trying to allocate projects before our intervention, and the problems it was experiencing with this process. The choice of approach we adopted for supporting their decision subsequently was an attempt of addressing these problems, which is discussed at the end of this section.

3.1. Attempting to define a general rule for allocation

The company started transferring projects from its UKDC to ADC at the beginning of 2004. At that time, the UK management team worked on identifying a general rule to distribute projects between the offices (e.g., client centred—transfer all work belonging to one client, module centred—transfer all work for one module, among others rules).

In their meetings, the above strategies were evaluated using Pros/Cons, with each rule receiving arbitrary “weighting”. It is not clear if the same criteria were used to evaluate each rule. The team took a long time to reach a decision and the result of the evaluation of strategies was to opt for transferring an entire module (module centred rule).

Project allocation decisions, however, did not follow the results of the semi-formal approach just described. The UK Project managers saw transfer to ADC as an immediate increase in the perceived risk to project success and a threat to their job security. In practice the decision of allocation was driven mostly by the consideration for UK team satisfaction and project risk reduction, ignoring the aspects listed in the formal process and without consideration of all stakeholders and conflicting objectives. This has resulted in various organisational problems, notably UK staff reluctance to change, their perception of lack of corporate strategic direction and of low competence of ADC. Aware of the strategic importance of this decision, and the damaging effects of the unsupported decision-making, the company asked for our help in supporting the process of project allocation.

3.2. Problems with the unsupported decision-making

Analysing the unsupported decision-making problem discussed above, we identified four main problems,
which are described below:

- **Poor quality decision-making**: The initial decision-making process, where they attempted to define a general rule for project allocation, focused on the software development properties and used attributes derived from them (interface touch points, algorithm complexity, etc.). The records of the meetings show that the assessment of listed attributes did not account for value trade-offs and were not employed consistently to evaluate the strategies. Even more critical, as an alternative-focused thinking attitude [10] it did not consider explicitly the strategic objectives of the organisation in the project allocation process (e.g. developing capabilities in the ADC).

- **Difficult to plan recurrent decisions**: The total number of projects scheduled for one release, and the type of projects, is forecast six months to a year in advance. This is used to plan the development capacity and team utilisation. However, the demand for certain skills, either business (knowledge of a module) or technical (programming language and module competence) is known very late and affects the long term planning for the off-shoring operation.

- **Lack of information sharing**: UK project managers know very little about the Asian office development and competencies, and about their corporate long-term strategy. This lack of information and collaboration experience influences their perception of ADC competence and of projects’ risk if a project is transferred. Even more critical, the lack of understanding of long term corporate strategy leads the project managers’ team to focus only on their own performance goals.

- **Uncertainty about performances**: Selection of projects for transfer to ADC considers software development projects at various stages of their life cycle and at various levels (entire project, or one activity or one task) to determine their “suitability” for transfer. The projects under investigation can be characterised by many properties whose values can be known with various levels of certainty depending on the project stage.

### 3.3. Selecting an approach for the intervention

These four main problems informed the choice of our approach to support their decisions. The issue of poor quality decision-making was addressed by the choice of a formal MCDA method, which allows a proper appraisal of alternatives and of value trade-offs. The recurrent nature of the decisions led us to propose the development of an MCDA-based decision support system. The lack of information sharing inspired us to use decision conferencing, an approach where the model is developed interactively with the group of decision makers [11]. This need for interaction with the clients also led us to use a multi-attribute value function model, due to its simplicity [12] and widespread use in decision conferencing. Finally, the problem of uncertainty about performances was addressed with the use of sensitivity analysis of performances on attributes when appraising projects with the MCDA model. The development of this model is described in the next section.

### 4. Developing the model

The decision support model was developed over a period of four months, working with the managers directly responsible for the projects’ outcome. This process was iterative and many of the phases overlap, as we attempted to represent in Fig. 1. This section describes the process of developing the model.

#### 4.1. The divergent phase

The first phase of the intervention focused on identifying the decision making group, scoping the decision situation and structuring the problem—this was the divergent phase, iteration cycles i1–i4, shown in Fig. 1. The researchers acted as process consultants, assuming that the clients did not know exactly what the problem was and needed help to define it. The aim was to construct a problem representation, interacting with the clients so they could learn more about their problem and also about their preferences [13].

#### 4.1.1. Framing the decision of software allocation

Initially, we attempted to frame the decision [10] through understanding the links between the top level strategy and the more detailed, organisational means objectives (Fig. 1, cycle i2). This analysis was conducted with the development manager, who was responsible for implementing the top level strategy through the development of the distributed team between ADC and UKDC. We collected information from the project managers (bottom-up in the organisation’s structure), and from the transition manager and internal strategy consultant (top-down), trying to understand their
decision-making process and identify the decision-makers group (iterations i2 and i3, Fig. 1). The outcome of this analysis is described below.

Prior to our intervention, the company set the strategic objectives as “reduce cost” and “increase capacity”. Fig. 2 depicts them at the plane 1, with a very large
corresponding frame of alternatives (represented as plane 5 in the same figure) i.e., all the means that allow reducing cost and increasing capacity. The company decided to develop a distributed team between two offices (rectangle A in plane 2) as a means of achieving those strategic objectives. Following this, as we described previously, a team of managers attempted to define a general rule for selecting, from the whole set of projects, those suitable to be sent to ADC (the small rectangle E in plane 5). As we discussed previously, such attempt was unsuccessful, in our opinion because of two main reasons: (i) it lacked an explicit objective of creating a strategy for developing the ADC; (ii) and it did not attempt to define a clear set of fundamental objectives [10] to evaluate the suitability of projects. Thus they were trying to use the rectangle A in plane 2 to choose, from all the projects (rectangle D) in plane 5, those suitable (rectangle E).

Therefore, our intervention focused on emphasizing the need of a strategy for developing the ADC (rectangle B in plane 3) and in defining a set of fundamental objectives (rectangle C in plane 4) which, at the same time, reflected such strategy and delimited the suitable projects to be sent to ADC (rectangle E in plane 5). The definition of these objectives was made through alternating between a value-focused thinking approach [10] (i.e., from the strategy of developing distributed teams—from rectangle B to rectangle C) and an alternative-focused thinking approach [10] (i.e., from projects’ attributes—from rectangle D to rectangle C).

4.1.2. Creating a value tree

Once the decision was framed, we started the process of developing a value tree with fundamental objectives. Resulting from a series of individual interviews with the project managers (Fig. 1, i3–i4) a value tree, decomposed into attributes and with multiple levels, was created. The managers agreed to continue work on criteria identification in group sessions. Meetings were scheduled in two week intervals and each one lasted approximately 2h. Each meeting resulted in modification of the value tree, for a total of five full versions. Its final version is displayed in Fig. 3.

4.2. The convergent phase

The convergent phase (Fig. 1, iterations i5–i7) consisted mainly of eliciting parameters of the multi-criteria model, testing the model with a sample of projects and checking its properties. These steps are described below.

4.2.1. Developing attributes and eliciting partial values for transfer

When the value tree was considered by the group as requisite [14] we started developing partial value functions (Fig. 1, i5). Through the process of eliciting the value functions, attributes were again modified, merged or decomposed, discarded and new ones added. Once the partial value functions and weights were identified, we introduced a draft decision support model in an Excel spreadsheet. As depicted in Fig. 3 there are six attributes in this model, five of which are grouped as “Risk attributes” and “Benefits attributes” (Table 1 describes them in detail).

These attributes have been assessed on scales determined with the group. The assessment values are then mapped to partial value for transfer functions, converting the assessment value to partial value for transfer. These functions were elicited from the group using the
<table>
<thead>
<tr>
<th>Attribute ⇒</th>
<th>Risk to product quality</th>
<th>Value for transfer</th>
<th>Impact of late delivery</th>
<th>Value for transfer</th>
<th>UK Priorities</th>
<th>Value for transfer</th>
<th>ADC training opportunities</th>
<th>Value for transfer</th>
<th>Strategic fit</th>
<th>Value for transfer</th>
<th>Days saved</th>
<th>Value for transfer</th>
<th>Project overall value for transfer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Notation</td>
<td>( q(a_i) )</td>
<td>( v_q(q(a_i)) )</td>
<td>( f(a_i) )</td>
<td>( v_f(f(a_i)) )</td>
<td>( p(a_i) )</td>
<td>( v_p(p(a_i)) )</td>
<td>( o(a_i) )</td>
<td>( v_o(o(a_i)) )</td>
<td>( S(p_i) )</td>
<td>( v_x(x(a_i)) )</td>
<td>( d(a_i) )</td>
<td>( v_d(d(a_i)) )</td>
<td>( V(a_i) )</td>
</tr>
</tbody>
</table>

**Assessment**
- Holistic assessment ranging from 100% to 0% considering requirement complexity, ADC capability, interface points, architecture, etc.
- Holistic judgement ranging from 100% to 0% considering estimated time for project completion, required dead line, possibility of delay and impact of the delay.
- Holistic assessment ranging from 100% to 0%, considering UK ownership, training opportunities and team utilisation.
- Holistic assessment ranging from 100% to 0%, measuring level of match with accepted strategy.
- Number of days saved for sending a project to ADC instead of doing in UKDC.

** maxx level**
- 100% 0
- 100% 0
- 100% 0
- 100% 100
- 100% 100
- 100% 100
- 150days 100 100

** Description max level**
- High possibility that project transfer to ADC will have adverse impact on product quality.
- High possibility of missing a fixed deadline with little or no lead time and having an adverse impact.
- Ideal training opportunity for UK or great attachment or ownership of work.
- Project provides high opportunity for training ADC team.
- Project area matches the current strategic goals.
- Saving 150 days or more.
- Project is ideal for transfer ADC.

** min level**
- 0% 100
- 0% 100
- 0% 0
- 0% 0
- 0% 0
- -20days 0 0

** Description min level**
- Project transfer to ADC will not increase risk to quality comparative to UK delivery.
- Flexible deadlines, lots of lead time, and low possibility of missing dead line (e.g. R&D).
- Routine job/working for UK team.
- Project is not suitable for training.
- Project area does not match current strategy.
- Investment 20 days or more.
- Project should be kept in UKDC.

**Weights**
- \( w_q = 0.28 \)
- \( w_f = 0.25 \)
- \( w_p = 0.17 \)
- \( w_o = 0.06 \)
- \( w_x = 0.03 \)
- \( w_d = 0.21 \)
4.2.2. Aggregating partial values for transfer

Each attribute had a set of reference levels defined, showing the best feasible and the worst acceptable situation. To aggregate the partial performance of a project, weights were assigned to each attribute. They were elicited using the swing weight procedure [12], anchoring the judgments on the worst and best levels of each attribute, firstly ranking the swings in order of preference and then assigning a value for each swing, considering the first swing as the reference.

Formally, each project (or group of activities) \( a_i \) is evaluated by a set of six attributes \( (q, t, p, o, s, d) \)—as described in Table 1. A partial value function \( v_j(.) \) is attached to each \( j \)th attribute, measuring the partial value of transferring \( a_i \) to ADC. Also, a weight \( w_j \) (see Table 1) is associated with each \( j \)th attribute, to convert the partial value of transfer into an overall scale. The overall value for transfer \( V(.) \) of a given project \( a_i \) was obtained via an additive model, a simple to understand and well-researched model for aggregating preferences [12]:

\[
V(a_i) = w_q v_q(q(a_i)) + w_t v_t(t(a_i)) + w_p v_p(p(a_i)) + w_o v_o(o(a_i)) + w_s v_s(s(a_i)) + w_d v_d(d(a_i)).
\]

The Excel workbook aided in adjusting model’s parameters (value functions and weights) and it was clear that the group preferred to work with a tangible model that could be manipulated. The group responded well to the mathematical concepts of the model (assessment of value trees and weights). Though the concept of swing weights was at first difficult to grasp, the
group developed quickly an appropriate understanding of it using the Excel model and seeing how weights expressed trade-offs.

4.2.3. Testing and implementing the model

Ten past projects were used to test the model throughout its development (Fig. 1, i6). The participants were asked to try to assess the past project on the decision criteria as at the time when the decision for, or against, transfer was made and use that as input to the model. Special attention was paid to the extreme cases, projects that were transferred and resulted in an adverse effect to the project success, or those that were considered for transfer but decided against.

While the group considered the results consistent and satisfactory, there was a concern that projects with high values in risk attributes needed to be highlighted. Testing the projects that were very close in their performance on other attributes and differ slightly on one of the risk attributes, both of which have heavy weights, the team recognised the need for some warnings about projects performing poorly on these criteria.

As a result, the model was enhanced with the creation of thresholds for some attribute and a switch flag for the entire project (see also [15]). Thresholds were assigned to two risk attribute setting the adequate levels of risk, and act as a communication flag. The switch turns on when the performance of a given project exceeds the threshold level (for example, if the perceived risk to product quality is above 70%). We also added a “manual” switch flag to account for extraordinary occurrences, such as lack of a key resource due to vacation schedule, which could prevent an otherwise well suited project from being transferred.

Through the value tree development iterations and use of the Excel workbook the model was tested for completeness, operationality, decomposability, absence of redundancy, conciseness and preferential independence (see [12] for details).

Acknowledging that any assessment provided in project post-mortem is always biased, as more information is available at the end of the project than it was at its beginning when the decisions were made, the group was satisfied with the model performance. Projects that were successfully delivered in the distributed team scored high in the overall value for transfer \( V(.) \) whilst those that were not successfully transferred, had a much lower result. Test results were used to “fine tune” the model, and also to build up client’s confidence in the model. The final model was formally presented to the client in our last meeting with them (Fig. 1, i7).

5. Conclusions and discussion

This paper described a real-world intervention, where a multiple criteria model was developed for supporting the allocation of software projects in distributed teams. It described in detail the iterative process of modelling, which tried to emphasise the importance of problem structuring and dynamicity in building up the MCDA model.

As an action-research (AR) intervention we were, at the same time, developing a MCDA-based decision support system and researching the impact of the intervention on the organisation (AR has been advocated as a powerful way of conducting research on group decision support systems [16] and increasingly has been employed in MCDA studies [17], not only in single but also in multiple interventions [18]). We collected qualitative data about the impact of the decision supporting (participant-observation during the intervention and a posteriori interviews with the decision-makers) and the conclusions that follow are based on these evidences.

At the beginning of our intervention we have identified some weaknesses of the unsupported decision-making process: poor quality decision-making, difficulties in planning recurrent decisions, lack of information sharing and uncertainty about performances. Through the process of problem structuring and model development, the decision situation seemed to be better understood by the stakeholders. This assisted in designing an operational decision procedure and the decision support model. The model provided basis for communication about project transfer issues and an accepted audit trail. In this manner, the implementation of the model seems to have improved efficiency of the decision-making process. The decision support model was accepted by the company as part of the standard project initiation procedure.

Through consideration of different decision frames and move between value-focused thinking and alternative-focused thinking, the management team discovered that not all the strategic implementation alternatives have been considered (i.e., there are other options available in plane 3—Fig. 2). This opened up areas for further exploration: following the intervention, new alternatives for strategy implementation focused on organisational structure were considered and a new implementation plan was created.

The process of model building seems to have improved organisational learning through improved inter-level communication and development of shared
This resulted in a better buy-in from project managers. This is consistent with the role of a decision analysis as a diagnostic tool and a collaborative, honest and truly interested in the research process. They were conscientious, collaborative, honest and truly interested in the research findings. They also quickly grasped the fundamentals of the methodology. Having a decision group with such high levels of commitment greatly improved the modelling process. Equally, the commitment from the upper management was necessary to see the process through to its completion.

Many times the information being disclosed was sensitive, there could be biased judgement or vested interest and it is crucial that the facilitators establish good rapport with the group. The decision makers needed to know that the results will not be used to “check on them”; although, at the end of our intervention, most of them asserted that the model was good for “audit trail” and decision justification.

5.1.1. Problem structuring was critical

The problem structuring phase was the longest one and it took us a while to identify and scope the decision context. This is consistent with the literature and research findings [12] in that the problems do not come neatly structured, and that clients often do not know what needs to be addressed, so the decision analyst acts as a process consultant. Therefore similar projects may need to take into account this feature, as well as the need for going back and redoing things as the decision makers learn more about their preferences and reflect about their problem.

5.1.2. The need for decision-makers’ commitment and trust on facilitators

The outlined process required an open attitude to dialog and negotiation, it was important to have a full commitment from the decision making group [12]. Project managers participating in this research displayed high level of professionalism. They were conscientious, collaborative, honest and truly interested in the research findings. They also quickly grasped the fundamentals of the methodology. Having a decision group with such high levels of commitment greatly improved the modelling process. Equally, the commitment from the upper management was necessary to see the process through to its completion.

Many times the information being disclosed was sensitive, there could be biased judgement or vested interest and it is crucial that the facilitators establish good rapport with the group. The decision makers needed to know that the results will not be used to “check on them”; although, at the end of our intervention, most of them asserted that the model was good for “audit trail” and decision justification.

5.1.3. The importance of elapsed time

The decision group thought that the process was very time consuming, nonetheless some believed the time invested was necessary and unavoidable. It should be noted that the time required to go through the process needs to account for the time to “absorb” new knowledge. As both, the decision analysis concepts and MCDA approach were new to the decision group, forcing the group to move along the learning curve faster would probably not work. The participants often confused the decision process with the decision criteria, with the project management issues, strategic direction and areas of responsibility. While it was difficult to separate the details, or other concerns, away from the decision criteria throughout the problem structuring phase, taking the time to achieve shared

5.1. Decision analysis modelling issues

Following the model acceptance, we conducted individual interviews with the participants to collect their impressions about the modelling process, its benefits and shortcomings. We also reflected on the intervention’s successes and challenges from our participant observation during the intervention. From this reflection, there were several issues highlighted by the case study, which could be of interest for future interventions using similar approaches and are discussed below.
understanding and separate problem layers enhanced the group field of vision and allowed the group to function. Once established, the level of required detail was not difficult to maintain. To assist in this process, it was very important to consistently use the same and precise terminology.

5.1.4. The need for interaction with decision-makers

It is also important to remember that the goal of the intervention was to design a functional decision support model that will be actively used. At the beginning of the intervention, many of the decision makers’ group expressed their scepticism in the applicability of any formal decision analysis, stating that their intuition will always override any results from a formal model. By designing a model through an iterative process and with maximum participation of decision makers, the facilitators aimed to create a model fully accepted by the decision makers group [14] as well as a model that would readily and easily be used. In this manner, the rich and prolonged interaction within the team, and with the model, enabled us to better understand and address the decision makers’ concerns.

Two previous conclusions, the importance of elapsed time and the interaction amongst participants and with the model, point to the learning that occurs through the modelling process as stated by Sterman: “In practice, effective learning from models occurs best, and perhaps only, when the decision makers participate actively in the development of the model” [20], p. 36.

5.1.5. Usefulness of an automatic DSS

The use of the Excel workbook was instrumental in facilitating the iterative development of the model. It is important to remember that the nature of this particular decision makers’ group: with software engineering background, the team was used to software applications and instant feedback from programming efforts. While developing a value tree and eliciting value functions was somewhat prolonged and challenging, as soon as the attributes and their partial value functions, were shown in a familiar format, similar to other software applications, the understanding of the modelling goal and the group’s response to the model increased dramatically. The team was actually excited to plug and play with the model, which greatly increased the participation, and, we believe, improved the model accuracy. Reflecting on this experience, we wonder whether introducing an Excel model earlier in the process would benefit the modelling efficiency.

5.1.6. Setting modelling goals from the beginning

Participants commented on the lack of clear modelling goals from the beginning of the process, which caused them a level of discomfort. We believe that introducing the goal earlier in the process, showing an example of a model from a similar situation, might had improved the modelling process efficiency. However, it could also work against creative problem structuring, thus reducing the alternatives and ideas generated [12]. In our opinion the choice of how focused is the decision process needs to take into account the balance between divergent and convergent thinking, to promote problem structuring and closure, respectively.

We conclude this set of reflections with a note of caution. While rich qualitative data about the impact of the intervention was collected, and compiled, in a systematic way, the data reflect merely facilitators’ and decision-makers’ perceptions—an inherent limitation of using any qualitative research methods. Therefore, our conclusions should be seen as merely tentative and open to further research. Future interventions in this type of problem may use also quantitative frameworks for assessing the benefits and challenges found in such kind of interventions, thus providing a means of triangulation of research results.

5.2. Directions for further model refining and future research

Applying a formal decision analysis method to the complex organisational issue uncovered inconsistencies in the interpretation of organisational and individual goals, and also provided the starting point for addressing these issues. Having accepted the decision support model at the current stage, the company intends to fine tune it during its use. It would be interesting to observe what changes in the model variables, preference values, weights and thresholds are made in response to changes in the decision context.

Another possible use of this ex-ante versus ex-post comparison would be the analysis of presumed consequences in the multi-criteria model, against the real outcomes when the product was developed. It may be the case that the task of selecting suitable projects to be sent to ADC would lead to better software engineering and, in this sense, the use of MCDA would be helping not only the decision of software engineers but also the process of software engineering (albeit indirectly). However we do recognise that this link between consequences and outcomes is a difficult relationship to assess, as external variables may influence such outcomes.
Using a multi-attribute value model was helpful in facilitating the forward movement during the group discussions. In addition to the decision support model, the organisation seems to have benefited from organisational learning, improved communication and uncovered issues. It would be interesting to investigate the results of other MCDA methods in the same context in terms of organisational and group outcomes.

The decision makers’ group was rather sceptical in the beginning of the process and expressed their belief in their “gut feeling” over the analytical approach. Despite being in agreement with the model at the conclusion of the research, some project managers still believe that their intuition will override the model’s advice. Further research could relate this attitude to their individual cognitive approaches, personality traits, and other individual preferences not accounted for in this model. It is possible that their reluctance to place their confidence in the formal model masks the fear of loss of organisational power due to the decision transparency that the model brings. However, the use of the model and the observation of results obtained could, in time, change the decision makers’ perception of the model applicability. A longitudinal follow-up research would deepen understanding of this research results, and could bring interesting results.

As a final remark, it is clear that MCDA is now a mature field with well-developed methods and many real-world interventions. However, we believe that more research focused on the process of supporting groups of decision-makers with MCDA is needed, in particular reflecting on the experience derived from these interventions (see also [17]). We hope that this paper may, somehow, further stimulate the debate about this important, and rather neglected, facet of the field.

Acknowledgement

We are thankful to two anonymous referees for their valuable suggestions; and to the decision-makers involved, who devoted their time in this intervention, and to the client company, for their support. This research was conducted when both authors were associated with Kingston Business School (London, UK).

References


