EVALUATION OF COTS USING MULTICRITERIA METHODOLOGY

Summary:

The quality software evaluation models present some specific characteristics: a hierarchical structure and evaluations at different abstraction levels, the notion of "presumed" evaluations, the mixture of different types of measures and of preferences requiring suitable aggregation procedures. In addition, the models are formalized by international standards largely used by practitioners.
In reality, applying the standards is difficult and the results are not completely satisfactory. The use of the multicriteria methodology allows to solve some of the problems. Such a methodology was experimented on several real cases already processed using the standard method. The results and learning of the experiment are presented in this paper, mainly: 1°) several evaluators are concerned, 2°) the quality model is refined during the evaluation process and therefore requires a validation procedure, 3°) it has to be meaningful both for each of the evaluators and theoretically, 4°) the aggregation procedures have to be adapted to the measures and to the evaluations and have to be integrated in the model.

Marie-José Blin and Alexis Tsoukiàs
Université Paris-Dauphine/Lamsade, Place du Maréchal de Lattre de Tassigny, 75775-Paris Cedex 16, France,
email: blin | tsoukias@lamsade.dauphine.fr, ph.: (0)1 44 05 47 25, (0)1 44 05 44 01, fax.(0)1 44 05 40 91

1. Introduction

Software quality is one of the most important enterprises' challenges of 2000. Firms are engaged in a cut-throat world competition and bringing a certification process into play allows to guarantee the skill of the firm or the conformity of a product, of a service or of an organization to a predefined reference.

No general certification of specific software exists, only certification of software relevant to a particular domain, for instance the critical software used in the aeronautical domain.

Concerning COTS, in France, the "NF Logiciel" label [AFNOR, 1996] certifies that*: the functions of the software match their description in the software documentation provided to the customer before purchasing, the software was tested by a registered laboratory, the software quality level is in accordance with the ISO 12119 standard requirements [ISO, 1994], the quality policy and practices of the supplier are verified, an after-sale service is provided and the software characteristics are durable.*

A great number of standards about software quality have been edited by ISO and IEEE. COTS quality evaluation process is defined by ISO 9126 [ISO 9126, 1991] and IEEE 1061 [IEEE, 1992] standards which propose to define the quality as a set of attributes, organized in a tree in which each attribute has a weight. The weighted sum is proposed to aggregate the measures. Besides, several authors tried to introduce multicriteria methodology in software evaluation [Zahedi, 1990], [Le Blanc, Jelassi, 1994], [Kontio, 1996], [Morisio, Tsoukiàs, 1997].

We studied three existing industrial cases of software evaluation processed in accordance with the standards and, faced with the difficulties to really exploit the results, we experimented the use of multicriteria methodology. This work allows us to understand some of the problems generated by the application of the standards, to propose new principles for evaluating software quality and to suggest future research for adapting multicriteria methodology to software quality evaluation.

The remaining of the paper is organized as follows: section 2 describes the studied cases and analyzes the difficulties to apply the standards to concrete evaluations of COSTS; section 3 discusses important general issues relevant to quality evaluation; section 4 explains how we applied the multicriteria methodology to the concrete cases and comments the results; section 5 concludes.

2. Using the standards to evaluate COTS

This section first presents the concrete cases which were used in the experiment. Secondly, the difficulties to use the standards are analyzed.

2.1. The evaluation problem

The cases studied belong to a French Laboratory which provides comparative studies of COTS and of hardware for publishing. Each of the cases involves six or seven software and two or three actors. They were processed in accordance with ISO 9126 standard using a five level hierarchical quality model. The different actors use the same hierarchy but give different weights to the elements. We worked only on a sub-tree of the quality model (the grey sub-tree of Figure 1) which contained from 200 to 300 leaves. The measures of the metrics may be: counts, Boolean or ranges of numerical scales. To normalize them, measures are transformed in marks by the formula below:

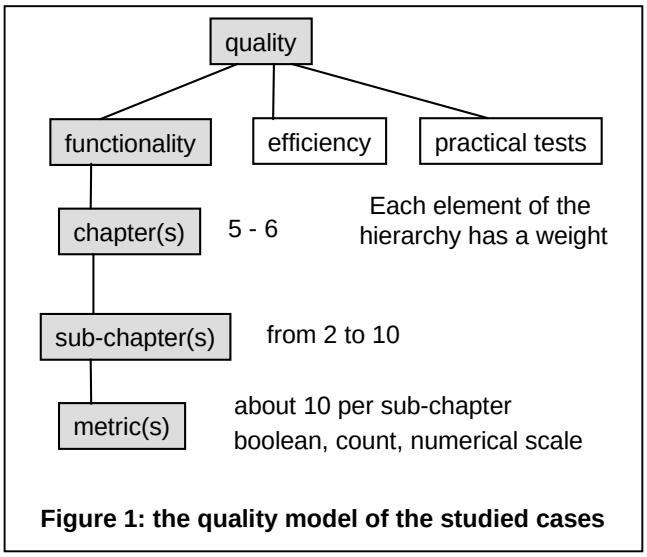

Figure 1: the quality model of the studied cases

mark = (measure / the highest measure of the metric) / (the sum of the weights of the metrics of the sub-chapter) * the weight of the metric

For example, Figure 2 represents the measures and their normalization of the metrics of the sub-chapter "Calendars" for two products A and B. Three metrics are defined: last year of the permanent calendar, maximum number of official holidays, definition of a specific calendar

of a task. The first metric is a count, the second one is a range of the numerical scale {0, 5, 10}, the last one is a Boolean.

metrics	type	weight	product A		product B	
			measure	mark	measure	mark
last year of the permanent calendar	count	1	2049	$\frac{2049}{2129} * \frac{1}{8}$	2129	$\frac{2129}{2129} * \frac{1}{8}$
maximum number of official holidays	range	2	5	$\frac{5}{10} * \frac{2}{8}$	10	$\frac{10}{10} * \frac{2}{8}$
definition of a specific calendar of a task	boolean	5	0	$\frac{0}{1} * \frac{5}{8}$	1	$\frac{1}{1} * \frac{5}{8}$

Figure 2: example of measure normalization in the studied cases

2.2. The difficulties to use the standards

In this paragraph, we make treasure of the difficulties encountered in the practical use of the standards. These results can be completed by the reading of [Fenton, Schneidewind, 1996] who discuss weak and strong points of standards.

COTS are generally largely used in the organizations which purchase them and it is impossible to define and to simulate all the applications the software will go through. Therefore, the evaluation has to consider mainly the software features rather than their behavior in a real context.

Evaluation of COTS is often a long process evolving in time and usually several actors are implied, as the final users, the purchase manager, the maintainers of the software, the manager responsible for the integration of the software in the organization or in the technical environment. Each of the actors has his own point of view and his own quality model. Several models have, often, common parts. Generally, each actor builds an a priori quality model with a great number of factors, sub-factors and criteria from his knowledge of domain and from his experience. But it is very difficult for him to determine the decisive elements of the model and to associate weights to factors, sub-factors and criteria. Moreover, the different elements forming a quality model are not always independent. For example, a same criterion may be associated with several factors resulting in an unfair distribution of the importance of different factors. Further on, quite often it happens that the evaluations associated to the leaves of the hierarchy are expressed in different types of scales including

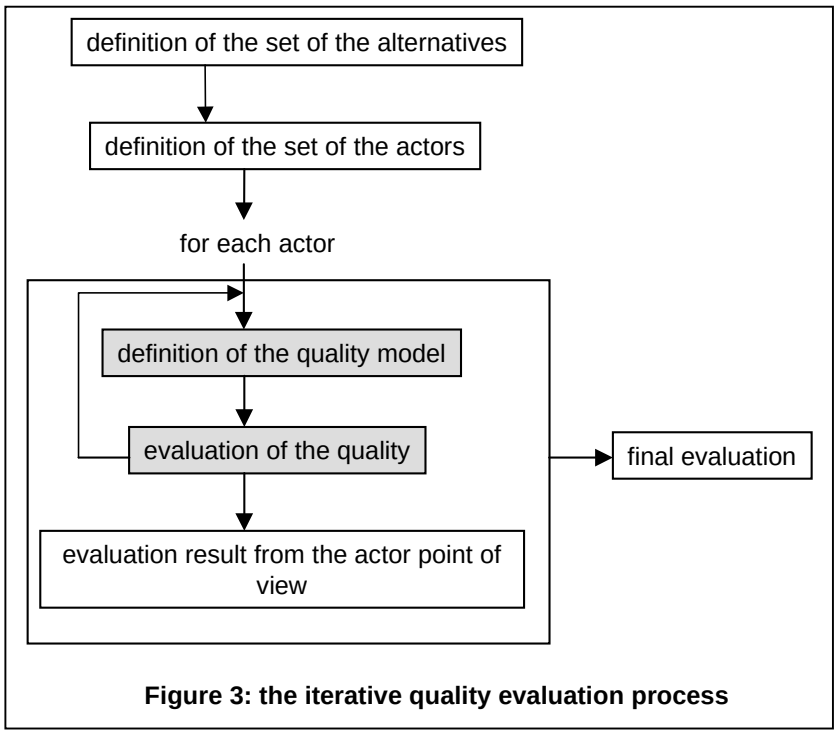

Figure 3: the iterative quality evaluation process

ordinal and nominal ones. Standards usually neglect the problem of adopting an appropriate aggregation procedure. In fact they do not consider the aggregation procedure as part of the quality model and this can result in totally unreliable and meaningless results.
The standards propose to transform the measures on the leaves of the hierarchy in homogeneous numerical evaluations (in order to use the weighted sum as aggregation procedure). However, such transformations are often arbitrary and the result is therefore meaningless. Some research have pointed out such a drawback (for a discussion see [Kitchenham, Pfleeger 1996]) on which we will come back in section 3.

Usually, the evaluators proceed by trial and error in order to determine the right choices. In fact, the quality model is defined during an iterative decision process, where the different elements are established by refining and adapting the original ideas (figure 3). But, the standards do not provide means to validate the model, to simplify and customize it.

In addition, a software is not a monolithic block but it is composed, bought and delivered in several parts depending on the needs of the users and it is complemented by services like assistance to its use or maintenance.

3. Further remarks concerning the aggregation problem.

This section discusses two important issues relevant to quality evaluation: the difference between a measure and an evaluation, and the drawbacks of aggregation. General recommendations for defining a quality model is then given.

3.1. Difference between a measure and an evaluation

The problem of quality evaluation (of a service, a product or whatsoever) is often addressed in a confusing way. The basic confusion arises between the "measurement" of quality and the choice (of an alternative) based on quality attributes. These are three completely different activities and have to be treated as such:

- the construction of a "measure" requires the definition of the semantics of the measure (what we measure?),
- the definition of the structure of the metric (what scale is used?),
- the definition of one or more standards (how the measure is performed?).

On the other hand, evaluating a set of alternatives under a decision perspective requires to answer questions of the type:

- who evaluates?
- why is the evaluation necessary?
- for what purpose is the evaluation?
- how the evaluation has to be done?
- who is responsible for the consequences?
- what resources are available for the evaluation?
- is there any uncertainty?

Finally, if for a given set A of alternatives, a measurement function exists, it is always possible to infer a preference relation from the measurement. However, such a preference relation is not unique (the fact that two objects have a different length, which is a measure, does not imply a precise preference among them). Suppose that $\exists\ l: A \rightarrow R$ (a measurement mapping the set A to reals, let us say lengths), then the following expressions are all admissible:

- $r(x,y) \Leftrightarrow l(x) = l(y)$
- $r(x,y) \Leftrightarrow l(x) = l(y)$

- $r(x,y) \Leftrightarrow l(x) = l(y)+k$
- $r(x,y) \Leftrightarrow l(x) = 2l(y)$, etc.

These are all admissible preference relations, but with an obvious different semantic. The choice of the "correct" one depends on the answers to the evaluation questions. An evaluation is therefore always a part of a decision aid process and represents its subjective dimension.

It is not always possible to obtain a measurement scale from a preference relation. First of all, the preference relation needs to be a complete binary relation (otherwise there is no guarantee that the numerical representation exists), but this is not always the case. Secondly, if the numerical representation exists, it is not necessarily unique. In such a case, it is difficult to choose the "correct" measure since we need to know all the possible sets on which such a measurement could apply. Finally, it is necessary to build an external metric and this may not be always possible.

3.2. Drawbacks of aggregation

Aggregating measures or preferences is a very common activity. Observations and/or evaluations provide measures or preferences on several distinct attributes or criteria. But we need a comprehensive measure or preference relation which may represent all the different dimensions we want to consider. It is surprising how often the choice of the aggregation operator is done without any critical consideration about its properties. Let us take two examples.

Example 1. Suppose one has two three dimension objects a, b, for which their dimensions are known: l(a), l(b), h(a), h(b), d(a), d(b). In order to have an aggregate measure of each object dimension, one may naturally compute their volume, that is $v(a) = l(a) * h(a) * d(a)$ and $v(b) = l(b) * h(b) * d(b)$. If the three dimensions are prices, one may use, however, an average, that is $p(a) = (l(a) + h(a) + d(a)) / 3$ and $p(b) = (l(b) + h(b) + d(b)) / 3$.
From a mathematical point of view, both operators are admissible when l(x), h(x), d(x) are ratio scales as in our example. However, the semantics of the two measures are quite different. It will make no sense to compute a geometric mean in order to have an idea of the price of a, b as it will make no sense to compute an arithmetic mean in order to have an idea of the dimension of a, b. The choice between the geometric and the arithmetic means depends on the semantics of the single measures and of the aggregated ones.

Example 2. Suppose one has two objects a, b and two criteria (in the Multicriteria Decision Aid Methodology, a criterion is a preference relation with a numerical representation) g_1 and g_2 such that, $\forall$ x, y, $p_j(x, y) \Leftrightarrow g_j(x) > g_j(y)$. Moreover $g_1: A \rightarrow [0, 1]$ and $g_1(a) = 0$ and $g_1(b) = 1$ and $g_2: A \rightarrow [0, 2]$ and $g_2(a) = 2$ and $g_2(b) = 1$. Under the hypothesis that the both criteria are of equal importance, many people will compute the average and infer the global preference relation. In our case, one has i(a, b) (i(x, y) representing indifference) since $g(a) = (g_1(a) + g_2(a)) / 2 = 1$ and $g(b) = (g_1(b) + g_2(b)) / 2 = 1$. However if an average is used, it is implicitly assumed that g_1 and g_2 admit ratio transformations. Therefore it is possible to replace g_2 by g_2': $A \rightarrow [0, 1]$ so that $g_2'(a) = 1$ and $g_2'(b) = 1/2$ (known as scale normalization). Under the usual hypothesis of equal importance of the two criteria, we obtain now p(b, a) since $g(a) = 1/2$ and $g(b) = 3/4$. Where is the problem?
The problem is that the average aggregation was chosen without verifying if the conditions under which, are admissible hold. First of all, if the values of a and b are obtained from ordinal evaluations (of the type good, medium, bad, etc.), then the numerical representation does not admit a ratio transformation (in other words we cannot use its cardinal information). Secondly, even if the ratio transformation was admissible, the concept of criteria importance

is misleading. In a "weighted arithmetic mean" (as the average is) the "weights" are constants representing the ratio between the evaluation scales. In the example, if we reduce g_2 to g_2', we have to give to g_2' twice the importance of g_1 in order to keep true the concept of "equal importance". In other words, it is not possible to speak about importance of the criteria (in the weighted arithmetic mean case) without considering the cardinality of their co-domains.

From the above examples, we can induce a simple rule. In order to choose appropriately an aggregation operator, it is necessary to take into consideration the semantics of the operator and of each single preference or measure and the properties (axiomatic) of the aggregation operator. In other words, if the aggregation operator is chosen randomly, neither the correctness of the result, nor its meaningfulness can be guaranteed. This is why we claim that the aggregation operator has to make part of the quality model.

3.3. General recommendations

As already discussed in the previous sections, software evaluation uses a complex hierarchical quality model. Moreover, the evaluation may concern parts of the software itself (or the whole), different dimensions and can be done for different purposes [Morisio, Tsoukiàs, 1997] and [Stamelos, Tsoukiàs, 1998].

From our discussion, it is clear that the definition of the measures, the criteria and the aggregation procedure cannot be done arbitrarily, but has to follow some general rules which we briefly outline in the following (the reader can see for more details [Morisio, Tsoukiàs, 1997] and [Blin, Tsoukiàs, 1998]).

Measures and criteria are usually inferred from the different points of view elaborated with the actors of the evaluation. Usually, such a set is obtained using international standards (as the ones included in ISO 9126 and IEEE 1061) and/or the actors' specific knowledge about the kind of the software to evaluate. Two processes are performed in a parallel way. The first, top-down, in which general dimensions (factors in the IEEE terminology) are desegregated to specific sub-dimensions and so on until sub-dimensions are reached on which the client is able to express or gather for or build up some information. The second, bottom-up, from the actors' specific knowledge who identifies subsets of evaluation dimensions as sub-dimensions of a dimension on a higher level. The result of the two processes is the definition of a hierarchy of the type presented in the previous sections.

However, in such an activity, it is necessary to pay attention not only to the semantic relevance of the son nodes of a parent node, but also in verifying their independence. The basic and necessary independence condition to meet is the "separability" of the "son-nodes" (the nodes to be aggregated in a parent node). Intuitively, the notion of separability means that if two objects are perfectly equivalent on all son-nodes except one, then the difference on such single son-node should be reflected to the parent node. In other words, every single son-node should be able to discriminate two objects alone. If such a condition is not verified, then the set of son-nodes has to be reconsidered. Further independence conditions can be imposed, but they deal with specific aggregation operators and will not be discussed here (see, however, [Roberts, 1979], [Von Winterfeldt, Edwards, 1986], [Vincke, 1992] and [Roy, 1996]).

For the definition of the aggregation operator associated to each node of the hierarchy (with the exception of the leaves) some basic rules can be remembered:

- if at least one of the numerical representations is obtained from an ordinal scale, then only ordinal aggregation operators can be used,

- if a linear multi-attribute value function is going to be used, then linear preferential independence on the set of criteria has to hold, a compensation principle is accepted and weights are trade-off,
- if an ordinal aggregation has to be used and a complete global preference relation is required, satisfying Pareto optimality either will be dictatorial, or will not respect the independence of irrelevant alternatives,
- a result of an aggregation cannot carry more information than the one contained in the aggregated nodes (for instance, it is not possible to construct a ratio scale aggregating ordinal scales),
- if the aggregation operator requires scale transformations, these have to be compatible with the admissible transformations of any single aggregated measurement (for instance, an interval transformation is not admissible on a ratio scale),
- if weighted statistics are used, weights should respect scale ratios (provided that such a ratio makes sense).

4. Applying the multicriteria methodology to evaluate COTS

Following the difficulties analyzed in the previous sections, we experiment the using of multicriteria methodology to the concrete evaluation cases presented in section 2. This section presents the experiment and its results.

4.1. Ordinal aggregation

An ordinal aggregation procedure belonging to the family of the ELECTRE methods [Roy, 1996] was used. We briefly summarize how the procedure works. The procedure provides a complete or partial ordering of equivalence classes from the best ones to the worst ones. It considers ties and incomparable classes. The procedure computes an ordering relation on all pairs of the alternatives set and constructs a preference relation on such a set. More precisely, for any pair of alternatives x, y, we have $S(x, y) \Leftrightarrow C(x, y) \wedge \neg D(x, y)$, where
S(x, y): "x is at least as good as y", C(x, y): a concordance condition which, in our case, is

$$C(x,y) \Leftrightarrow \frac{\sum_{j \in J_{xy}^{\succeq}} w_j}{\sum w_j} \geq c \quad and \quad \frac{\sum_{j \in J_{xy}^{\succ}} w_j}{\sum_{j \in J_{xy}^{\prec}} w_j} \geq 1$$

with $J_{xy}^{\succeq}$ the set of criteria for which $S_j(x, y)$ holds, $J_{xy}^{\succ}$ the set of criteria for which $\neg S_j(y, x)$ holds and $J_{xy}^{\prec}$ the set of criteria for which $\neg S_j(x, y)$ holds.

$D(x, y) \Leftrightarrow \exists g_j: v_j(x, y)$ where:
$v_j(x, y)$ is a veto condition on criterion g_j holding, for instance, if $g_j(y) \geq g_j(x) + v_j$, v_j being a threshold.
The conditions under which $S_j(x, y)$ holds depend on the preference model associated to criterion g_j. Once a global relation S (x, y) is obtained, it is possible to establish:

- a strict preference relation $P(x, y) \Leftrightarrow S(x, y) \wedge \neg S(y, x)$;
- an indifference relation $I(x, y) \Leftrightarrow S(x, y) \wedge S(y, x)$;
- an incomparability relation $R(x, y) \Leftrightarrow \neg S(x, y) \wedge \neg S(y, x)$.

The procedure was applied from the leaves to the root of the quality model. At each level of the tree, an ordering of the alternatives (the software to evaluate) is calculated for each node of

the level. These orderings are used at the next level to calculate new orderings and so on (aggregation of preferences) (figure 4).

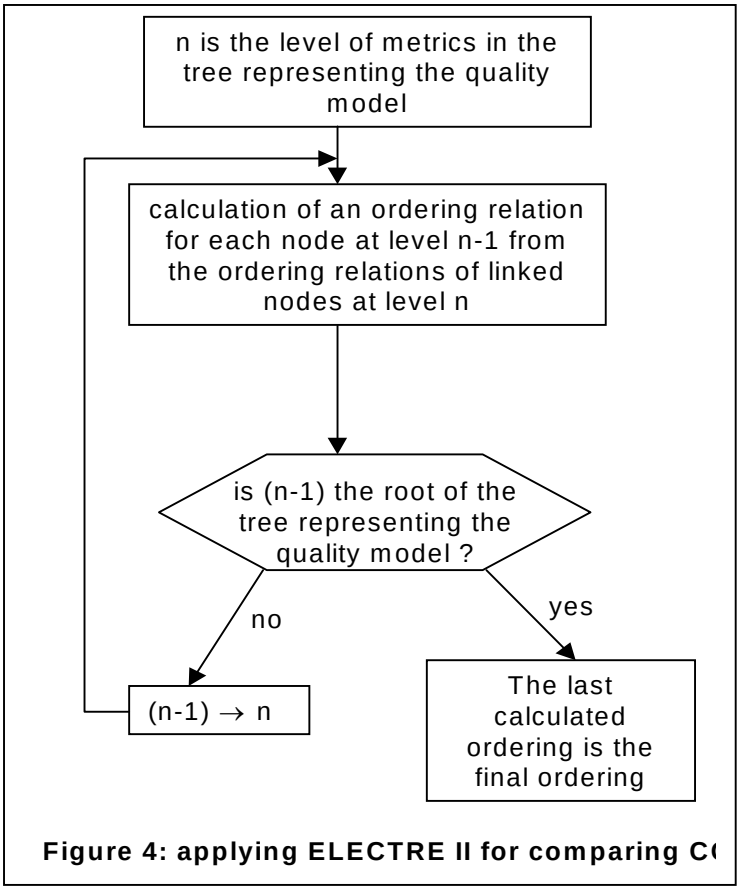

Figure 4: applying ELECTRE II for comparing C(

The concordance formula was only used. The responsible of the evaluation in the Laboratory was not able to indicate any veto condition on the criteria. Moreover, as most of the criteria were ordinal, it was very difficult to state any veto threshold. Finally, the responsible considered that the existence of a veto could act as an a priori elimination, in which case, the set of products to evaluate should be considered as badly chosen.

In order to be able to repeat the calculation at each level of the quality model, the outranking relation obtained at each node of the hierarchy was transformed into a weak order using the "Score method". This method consists, for each alternative, in subtracting the number of times this alternative outranks the others and the number of times it is outranked ("final score" of each alternative) (figure 5). The weak order is established on the basis of the final score of each alternative.

	A1	A2	A3	A4	# of alternatives outranked by *x*
A1	1	0	0	0	1
A2	1	1	1	1	4
A3	1	0	1	1	3
A4	1	0	0	1	2
# of alternatives outranking *x*	4	1	2	3	
final score	-3	3	1	-1	
final order used in the next level of hierarchy	4	1	2	3	1 being the best

Figure 5: using the Score method to order alternatives at each node of the quality model

The ordinal aggregation may conceal situations of incomparability which have to be analyzed before calculating the final order of the alternatives at each level of the hierarchy. When incomparable alternatives were detected, a sensitivity analysis was applied. Every alternative better or worst than the incomparable alternatives was kept away. The incomparable alternatives and every alternatives ordered between them were retained and the calculation were repeated with a new concordance threshold until all incomparabilities disappeared. The idea of the sensitivity analysis is to verify at what confidence level all the alternatives can be compared. In fact, the decision maker wanted to verify if the incomparability was due to the imposition of high confidence or to intrinsic characteristics of the alternatives.

4.2. Geometric means

Another experiment was conducted using the geometric and dual geometric means as aggregation procedures. For that, we considered the values on the leaves of the hierarchy as an evaluation of " attractiveness " expressed on the scale [0, 1]. More specifically, the two formulas below were used:

$$u(x) = \Pi_j (u_j(x))^{w_j} \qquad \text{geometric mean}$$

$$u(x) = 1 - \Pi_j (1- u_j(x))^{w_j} \qquad \text{dual geometric mean}$$

where:

- u(x): score of alternative x on the parent node,
- $u_j(x)$: score of alternative x on the son node j,
- w_j: relative importance of the son node j.

We avoided the extreme values 0 and 1 since the presence of just one of them in the son nodes will keep the global score to 0 or 1 independently of the rest of the evaluations (in other words we attenuated the non compensation effect of the formula).

4.3. Comments on the experiment

The use of the ordinal aggregation methods presented two positive features and a negative one:

1. it enables to handle homogeneously non homogeneous information in a meaningful way since it does not impose any restriction on the information expressed on the criteria (sub-criteria, etc.).
2. it enables to put in evidence situations of incomparability which otherwise could be concealed during the aggregation. Therefore, the ordinal aggregation can be a way to validate the quality model.
3. the information contained in each criterion (sub-criterion, etc.) is often richer than the simple order of the alternatives. It is sometimes a ratio or interval information on the comparison of the alternatives, other times an external measurement or a qualitative judgment, but in all such cases, it contains knowledge about a metric and its properties. A purely ordinal aggregation in every level eliminates these information since it focuses on the order of the alternatives. This may lead to a poor conclusion from the point of view of the decision maker. Particularly, in our case, although the client was aware that a large part of his criteria was purely ordinal, the decision maker would like to have measures of the distances between the alternatives.

Geometric mean brings out specific "bad" performances of the alternatives since the global score deteriorates exponentially with respect to the importance of the criterion on which the "bad" score is expressed (conversely the dual geometric mean will bring out alternatives with "good" evaluations). Under such a perspective, both means introduce a non linear compensation effect among the criteria and therefore can be used as measures of "attractiveness" in the interval [0, 1] in the presence of ordinal information also. They may also be replaced with other kinds of ordered statistics.

5.Conclusion

The paper addresses the problem of software evaluation mainly as far as COTS are concerned. The use of the multicriteria methodology is advocated as a general framework under which both the problem of defining a client meaningful model and a theoretically sound model can be addressed. In the paper, we discuss the problem of how international standards conceive software evaluation and why such an approach can be misleading and dangerous if not associated to a more rigorous methodological framework. Some drawbacks of inaccurate

choice of aggregation procedures within quality models are presented and discussed besides further features of quality models specific to the software domain. An experiment, using ordinal multicriteria aggregation methods is presented and the lessons learned are discussed. Future research concerns the use of ordinal measurement procedures derived from sorting multicriteria methods and the introduction of validation procedures in the process of quality model definition.

References

Association Française de NORmalisation: ***Règlement NF Logiciel***, Edition 1.0, 8 mai 1996

MJ. Blin, A. Tsoukiàs: ***Multicriteria Methodology Contribution to the Software Quality Evaluation,*** Lamsade/Université Paris-Dauphine, Document no. 155, mai 1998

Norman Fenton, Norman F. Schneidewind***, Do Standards Improve Quality?,*** IEEE Software, pages 22 - 24, January 1996

The Institute of Electrical and Electronics Engineers: ***Standard for a Software Quality Metrics Methodology***, December 1992

International Organization for Standardization***: ISO 9126: Information Technology - Software product evaluation - Quality characteristics and guidelines for their use***, 1991
International Organization for Standardization, ***ISO 12119: Information Technology - Software, Package, Quality Requirements and Testing***, 1994

Barbara Kitchenham, Shari Lawrence Pfleeger: ***Software Quality: The Elusive Target***, IEEE Software, pages 12 - 21, January 1996

J. Kontio: ***A Case Study in Applying a Systematic Method for COTS Selection,*** Proceedings of the 18th ICSE, pages 201 - 209, 1996

Louis le Blanc, Tawfik Jelassi: ***An empirical assessment of choice models for software selection: a comparison of the LWA and MAUT techniques***, Revue des systèmes de décision vol. 3 - no.2, pages 115 - 126, 1994

M. Morisio, A.Tsoukiàs***: IusWare: a methodology for the evaluation and selection of software products***, IEE.-Softw. Eng. Vol. 144, pages 162 - 174, 1997

Roberts F.S.: ***Measurement Theory with Applications to Decision Making, Utility and the Social Sciences,*** Addison Wesley, New York, 1979

Roy B.: ***Multicriteria Methodology for Decision Aiding***, Kluwer Academic Publishers, Dordrecht, ISBN 0-7923-4166-X, 1996

Stamelos J., Tsoukiàs: ***Software Evaluation Problem Situations***, Lamsade/Université Paris-Dauphine, Document no. 156, 1998

Vincke Ph.: ***Exploitation of a crisp relation in a ranking problem***, Theory and Decision, vol. 32, pages 221-240, 1992

Winterfeldt Von D., Edwards W.: ***Decision Analysis and Behavioral Research,*** Cambridge University Press, Cambridge MA., 1986

Zahedi F: ***A method for quantitative evaluation of expert systems***, European Journal of Operational Research, vol. 48, 136 – 147, 1990