

A Model for Choosing the Right ERP System

Aristides Dasso, Ana Funes

Universidad Nacional de San Luis, Ejército de Los Andes 950
D5700HWZ San Luis, Argentina
{arisdas, afunes}@unsl.edu.ar

Abstract. This work presents a model for the evaluation and selection of Enterprise Resource Planning (ERP) systems. Although these systems are used more and more there have been reports of different kind of problems, underlining the need of having models for their evaluation. Evaluation also permits the possibility, for the prospective user, of being able to establish its own requirements and compare different systems to assess which is the one that better adapts to the user's needs. The model we present here is based on the Logic Score of Preference (LSP) method. An overview of the LSP method is also given.

Keywords: Enterprise Resource Planning. ERP. System Evaluation methods. Continuous Logic. LSP Method.

1 Introduction

In their well-known report [10], Escalle et al give a useful description of what an Enterprise Resource Planning system is; they establish that “ERP, when successfully implemented, links financial, manufacturing, human resources, distribution, and order management systems into a tightly integrated single system with shared data and visibility across the business.”

Several people and organizations started using ERP in the 1990's as a way of referring to the integrated system that organizations use to conduct all of its business processes. The idea behind it is that all the different systems –payroll, accounting, stock management, etc.– used in an organization could be integrated into one system.

Many organizations of all kind –government, industry, retail, etc.– have installed ERP systems with the clear goal of improving their business processes.

The usefulness of ERP is widely recognized, and one example of this is that ERP industry is a multibillion-dollar business.

However, one key point in Escalle et al's description given above is the phrase “when successfully implemented”. This is a very important condition given that not all the implementations of ERP are a success story. On the contrary, many stories of failures in the implementations of ERP systems abound.

Most of the causes for these failures are attributed to errors committed in the initial faces of the implementation, and precisely in choosing the right ERP and the correct characteristics of the systems for the organization [22]. As Maya Daneva establishes

in [13] “Given that RE (requirements engineering) is any ERP project’s most expensive stage, this knowledge is not only needed but also vital to the field.”

In addition, many ERP vendors offer guides (e.g. [9], [11], [16]) for prospective clients to help them choose not only the right ERP system but also the correct configuration that will best serve the client’s needs. Furthermore, there are a number of organizations such as TEC Technology Evaluation Centers [20], SpecIT [17] that are in the business of advising future ERP users or those that want to change or upgrade their ERP systems and they normally have methods for choosing systems.

There are in the literature a number of recommendations for choosing and evaluating ERP. In [23] Franch and Carvallo concern themselves mainly with the quality of ERP systems and use the “International Organization for Standardization and International Electrotechnical Commission 9126-1 quality standard” to build their model. The paper also gives a number of works related to the construction of quality models. Ufuk Cebeci [21] employs Analytic Hierarchy Process (AHP) combined with Balanced Scorecard (BSC) to create a model for the selection of ERP system for the textile industry. Others use a combination of traditional techniques, e.g.: Avram et al [2] they “explain(s) ways of creating efficient ERP strategies based on the entire ERP lifecycle, from business analysis, process engineering, system analysis and design, implementation and maintenance support, and focusing on the organization’s strategic processes”. Wang et al [19] employ Incomplete Linguistic Preference Relations coupled to a variation of AHP to evaluate ERP system suppliers. Wei et al [1] is nearly a classic example of using AHP to select an ERP. Onut and Efendigil [15] use Fuzzy AHP for their model to evaluate ERP systems. Also Ya-Yueh Shih [24] adopts Fuzzy AHP for its model. In [12] Jafar Razmi and Mohamad Sadegh Sangari apply a multiple-criteria decision analysis (MCDA), i.e. a technique for order of preference by similarity to ideal solution (TOPSIS) and use PROMETHEE [14] as a tool to create their model.

In this work, we present a model for the evaluation and selection of ERP systems. The model has been developed following the Logic Score of Preference (LSP) method [8], [6], [5], [7] –a method that allows the creation of models for the evaluation, comparison and selection of complex systems. It prescribes a number of steps to be followed and it provides a number of operators of a Continuous Logic that combined in the right way permits the development of evaluation models according to the user requirements.

The rest of this work is organized as follows. Section 2 presents an overview of ERP systems while a brief description of the LSP method is presented in Section 3, both necessary for the reading of the work. In Section 4, we present our model showing part of the identified requirement tree, some elementary criteria as well as part of the developed aggregation structure. Finally, in Section 5, we close the work with some conclusions and future work.

2 Enterprise Resource Planning Systems

ERP systems give support from small to big enterprises in the integration of their business processes –planning, manufacturing, etc.–, into one automated system.

The main advantage in adopting an ERP system is that integrating businesses processes saves time and money. Management can make decisions faster and with fewer errors and data becomes visible across the organization.

ERP systems normally offer a number of modules, supporting the different business processes, which can be tailored to the specific needs of a given organization.

They automate the organization's activities within an integrated software application. In this way, information can flow between business functions inside the organization at the same time that ERP allows interaction with outside stakeholders.

Several software corporations such as SAP, Oracle, Microsoft, Infor, among many others, offer different products at the same time that they provide advice in choosing the software solution that best fit the business as well as advising in the configuration and deployment of the ERP system.

Choosing the right ERP system for a particular organization is not an easy task. It implies the selection of the right set of features to make sure that those chosen do not fall short or include unnecessary ones, therefore increasing costs as well as hindering the business processes instead of enhancing them.

When organizations make system selection without supporting their decision following a formal methodology, usually this leads to a poor choice. This is often due to consider an incomplete or a bias list of requirements, or because management organization has relied too much on vendor demos or it has postponed important aspects such as functionality or infrastructure in favour of price.

3 The LSP Method

The Logic Scoring of Preference (LSP) [8], [6], [5], [7] method we have applied in the development of the model presented here is a method employed for the realization of complex criterion functions and their application in the evaluation, optimization, comparison and selection of general complex systems.

The LSP method can be used to evaluate complex systems and, since it is a general evaluation method, it can also be employed in particular in the evaluation processes involved in the choosing of an ERP system.

Since this method is not a simple additive scoring method but allows the use of complex and/or decisions, is especially useful where these conditions apply.

As a starting point in LSP, it must be clearly determined what are the user requirements, the main attributes of the system and their value preferences. These attributes are called *performance variables*. Each one of these variables is mapped into an *elementary preference* by defining and applying the corresponding *elementary criteria*.

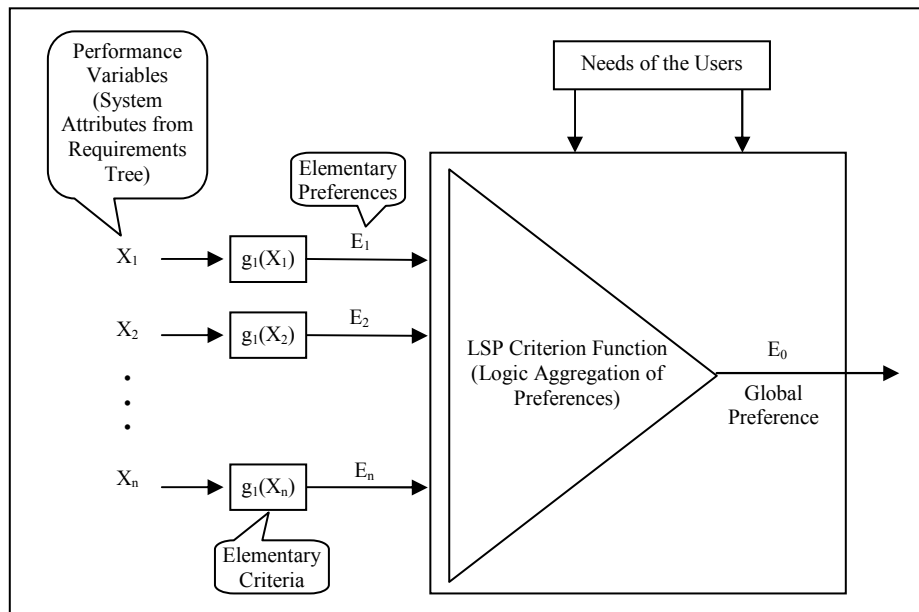


Figure 1. An overview of the LSP evaluation process.

An elementary criterion is a function that transforms a real value, coming from a performance variable, into a value belonging to the $[0,100]$ interval –i.e. the corresponding elementary preference. An elementary preference represents the degree of fulfilment of a requirement, where 0 means that the requirement has not been fulfilled at all and 100 that it has been completely satisfied. Therefore, to define an elementary criterion it is necessary to have some previous experience to determine what the acceptable range of values is for the corresponding performance variable.

The elementary preferences, obtained from the transformation of the performance variables via the corresponding elementary criteria, are used as input to the *LSP criterion function*.

The LSP criterion function or *Aggregation Structure* is a function that yields a single global indicator between 0 and 100 of the degree of fulfilment of the whole system requirements. It is built by aggregating the elementary preferences by means of a set of operators. To aggregate preferences means to replace a group of preferences (the input preferences) by a single preference (the output preference), which denotes the degree of satisfaction of the evaluator with respect to the whole group of input preferences. The output preferences must be aggregated again until a single global preference is obtained.

To calibrate the LSP criterion function it is necessary to take into account the needs of the end users. The process of calibration obviously represents the most complex phase in the whole evaluation. The global preference –obtained as output of the LSP function– is the result of the combination of the elementary preferences taking into account both the relative importance of each preference and the necessary logic relationship between them.

Once the calibration of the LSP criterion function has finished, the evaluation of

each competitive system can start. It means that, for each competitive system, we must provide as input to the aggregation model the set of values corresponding to the performance variables to obtain as output a global performance indicator for each system under evaluation.

An overview of the applied method is shown in Figure 1, where the performance variables X_1, \dots, X_n from the system requirement tree are transformed, by means of the elementary criteria g_1, \dots, g_n , into the elementary preferences E_1, \dots, E_n . These elementary preferences are provided as input to the aggregation structure or LSP criterion function, which returns a single global indicator E_0 representing the fulfilment degree of all the requirements of the system under evaluation.

4 A Model for Choosing an ERP System

In this section we present the result of following the steps proposed by the LSP method to create our final evaluation model (or LSP criterion function), which reflects the requirements an ERP system must fulfil and yields a quantitative indicator of the requirement satisfaction level (global preference).

The main activities achieved to obtain the model –development of the system requirement tree, definition of elementary criteria and aggregation of preferences– are described in Sections 4.1, 4.2 and 4.3.

4.1 System Requirement Tree

In order to develop an exhaustive list of requirements, we have applied a hierarchical decomposition process for requirement derivation. At the beginning, we defined all major groups of requirements, and then through successive decompositions we decomposed each group into subgroups. By repeating this process, we obtained our *system requirement tree*, whose leaves correspond to the performance variables.

Table 1. Requirement tree showing only its first level.

1. Budgeting
2. Costing
3. Billing
4. Project Management
5. Resource Planning
6. Scheduling
7. Opportunity, Contact, and Contract Management
8. Time and Expense Management
9. Knowledge Management
10. Third Party Integration
11. Internal Office Functionality
12. Product Technology

It must be noted that the requirement tree can be as big as the evaluator desires. It can have only a few entries or hundreds or even thousands. Table 1 shows the first

level of our requirement tree for the evaluation of ERP systems. We have constructed the first level of the requirement tree having in mind the main and most common requirements for an ERP system; let us note that, as it was said before, the tree can be as detailed as necessary.

For reasons of space, we show here only part of the model. We have chosen to illustrate our model considering only a part of the requirement tree, which is shown in Table 2, and that corresponds to the expansion of item 2 from Table 1.

Let us bear in mind that an ERP system can have more than a thousand items, and although some of them can be, and shall be, specific to a particular organization, e.g. manufacturing, we have only used for our example a few of the most common requirements. The evaluator looking to choose an ERP system for a specific business will have to construct his/her own requirement tree with the features that pertain his/her own organization business processes.

Table 2. Expansion of item 2 “Costing” in the requirement tree of Table 1.

2. Costing
2.1. Project Costing
2.1.1. Expenditures
2.1.1.1. Fixed costs
2.1.1.2. Variable costs
2.1.1.3. Definition of expenditure categories
2.1.1.4. Definition of expenditure types
2.1.1.5. Expenditures for import or export
2.1.1.6. Multiple currency transactions
2.1.1.7. Flexible labour rates
2.1.1.8. Rule-based accounting
2.1.1.9. Automatic calculation of actual cost at the task and project levels
2.1.1.10. Forecast cost at completion
2.1.1.11. Updating of project costs based on
2.1.1.11.1. actual time
2.1.1.11.2. time to complete
2.1.1.12. Definable cost codes
2.1.1.13. Management of labour and non-labour costs
2.1.1.14. Management of the recharge of non-labour costs to clients
2.1.1.15. Estimation of project costs, or fixed costs, or both
2.1.1.16. Capital versus current expense
2.1.2. Allocations and Burdening
2.1.2.1. Assignment and definition of burden multipliers
2.1.2.2. Definition of burden versions and schedule
2.1.2.3. Revision and override of burden schedules
2.1.2.4. Definition of allocation rules
2.1.2.5. Cost distribution of allocations
2.1.2.6. Process allocations runs
2.1.2.7. Unlimited definition of cost items allocated to the organization tree
2.1.2.8. Chargeback allocations
2.1.3. Project Costing Adjustments
2.1.3.1. Multiple types of expenditure adjustments
2.1.3.2. Mass adjustments
2.1.3.3. Adjustment of multicurrency transactions
2.1.3.4. On-line processing and review of adjustments

Table 2. Expansion of item 2 “Costing” in the requirement tree of Table 1.

2.1.3.5. Audit trail for expenditure adjustments
2.1.3.6. Billing

4.2 Definition of Elementary Criteria

Once we developed the requirement tree and determined the performance variables, we started with the definition of the elementary criteria.

An elementary criterion is a mapping from a performance variable value into an elementary preference value. Since the interpretation of an elementary preference is the degree of fulfilment of a given requirement, this is a real number belonging to the interval $[0, 100]$. Therefore, the value 0 corresponds to a situation where the performance variable does not satisfy the requirements and the value 100 to the case that the requirement is completely fulfilled, while the values between 0 and 100 denote partial satisfaction of requirements.

Given that only the leaves of the requirement tree correspond to performance variables, the elementary criteria must be defined only for the requirement tree’s leaves.

Some of the elementary criteria we have defined for obtaining the elementary preferences in our model are shown in Table 3. The table shows the elementary criteria defined for items 2.1.3.1. “Multiple types of expenditure adjustments” and 2.1.3.2. “Mass adjustments”. The examples shown illustrate two of the very many elementary criteria possible. For a more complete understanding of Table 3 we refer the reader to the explanation given in [3] – [8].

Table 3. Some elementary criteria for item 2.1.3

2.1.3. Project Costing Adjustments
2.1.3.1. Multiple types of expenditure adjustments (TE)
$g(TE) = \begin{cases} 100 & \text{if } TTE > 10 \\ TTE * 10 & \text{if } TTE \leq 10 \end{cases}$
where TTE : Total Types of Expenditures
2.1.3.2. Mass Adjustments (MA)
$g(MA) = \begin{cases} 100 & \text{if } MA = \text{Excellent} \\ 80 & \text{if } MA = \text{Very Good} \\ 70 & \text{if } MA = \text{Good} \\ 40 & \text{if } MA = \text{Fair} \\ 20 & \text{if } MA = \text{Poor} \\ 0 & \text{if } MA = \text{Very Poor} \end{cases}$

4.3 Aggregation of Preferences

Once the requirement tree has been finished and the elementary criteria defined, we can start with the aggregation of preferences. This process uses the structure of the

system requirement tree to build a new tree structure, the final aggregation structure or LSP criterion function.

The process starts by aggregating groups of related elementary preferences and generating, in this way, subsystem preferences. Therefore, the elementary preferences, corresponding to the requirement tree leaves, are aggregated in new preferences. This bottom up process is repeated with the resulting groups of preferences until a single global preference can be obtained. The logic aggregation structure created by applying the described process must reflect the user requirements that in this case are the requirements for an ERP system.

If we want to aggregate n elementary preferences E_1, \dots, E_n in a single preference E , the resulting preference E –interpreted as the degree of satisfaction of the n requirements– is expressed by a function having the following properties:

1. The relative importance of each elementary preference E_i ($i= 1..n$) is expressed by a weight W_i ,
2. $\min(E_1, \dots, E_n) \leq E \leq \max(E_1, \dots, E_n)$.

These functions are obtained from the instantiation of the weighted power means:

$$E(r) = (W_1 E_1^r + W_2 E_2^r + \dots + W_n E_n^r)^{1/r}, \text{ where}$$

$$0 < W_i < 100, 0 \leq E_i \leq 100, i = 1, \dots, n, W_1 + \dots + W_n = 1, -\infty \leq r \leq +\infty$$

The choice of r determines the location of $E(r)$ between the minimum value $E_{\min} = \min(E_1, \dots, E_n)$ and the maximum value $E_{\max} = \max(E_1, \dots, E_n)$. For $r = -\infty$ the weighted power mean reduces to the pure conjunction (the minimum function) and for $r = +\infty$ to the pure disjunction (the maximum function), giving place to a Continuous Preference Logic (CPL). The range between pure conjunction and pure disjunction is usually covered by a sequence of equidistantly located CPL operators named: C, C++, C+, C+, CA, C-, C-, C-, A, D-, D-, D-, D+, DA, D+, D+, D+, D+. For a more detailed description of the technique for selection of r see [7] and [3].

The weights associated to each elementary preference are assigned by the user according to the importance that each elementary preference has in the model being constructed. The same goes when choosing the different CPL operators.

In Figure 2 to Figure 5 we show the aggregation structures we have built for some of the items in the requirement tree given in Table 2. In the depicted figures, circles represent CPL operators –also referred as Generalized Conjunction Disjunction (GCD) operators, rectangles correspond to elementary preferences and the weights are shown over the edges. Rounded rectangles in light grey do not form part of the aggregation structure. We have introduced them to indicate partial preferences corresponding to the aggregation of set of items in the requirement tree.

In the aggregation structure shown in Figure 2, all input requirements are considered as non-mandatory, i.e. if any one of them is missing (its value is zero) then the resulting preference will be not necessarily equal to zero. We have made this decision since these are items that refer to performance values in the requirement tree that are desirable but non-mandatory.

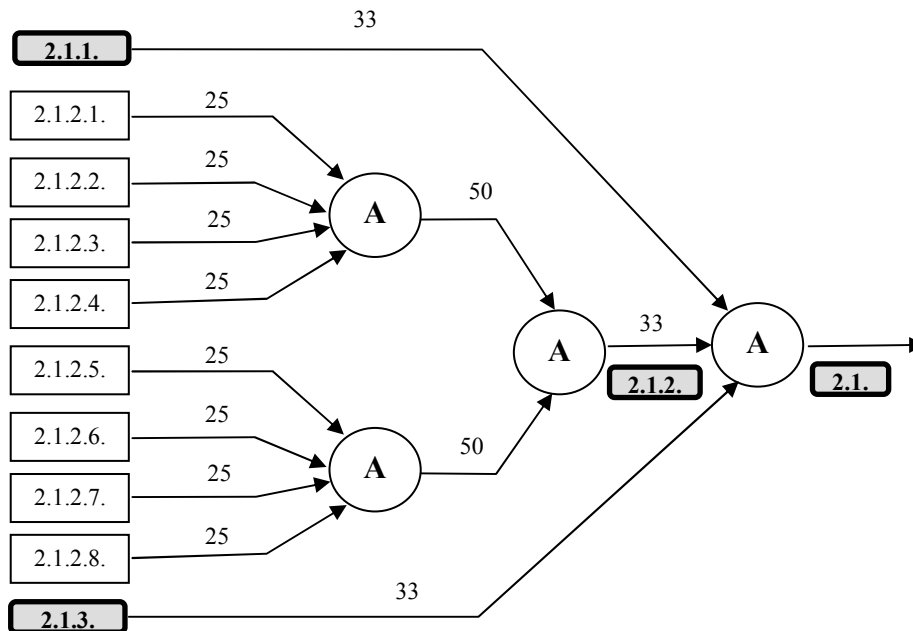


Figure 2. Aggregation Structure for item 2.1. “Project Costing”.

This is also the reason for the choice of the **A** operator that allows to compute the arithmetic media of the input preferences. The eight elementary preferences that form item 2.1.2. “Allocations and Burdening” have been aggregated in two groups of four preferences each and we have applied the same operator **A** three times (once for each group of four items and once for the aggregation of both). This is so because the tool that supports the aggregation structure edition has the CPL operators calculated for two to five parameters. For reasons of space, we do not show the structures for items 2.1.1 “Expenditures” and 2.1.3. “Project Costing Adjustments”, whose preferences have been aggregated by the **A** operator with the resulting preference 2.1.2 and that are represented in the figure by the light grey rounded rectangles 2.1.1., 2.1.2. and 2.1.3.

In Figure 3, we show the elementary preferences for item 2.1.1.11. “Updating of project costs based on”. The choice of the CPL operators for the structure shown in this figure follows a strict policy, i.e. we have considered both elementary preferences in item 2.1.1.11 are essential, namely, none of them may be missing (its value is zero) otherwise the resulting preference will be zero. This sub structure will be rejecting, at least partially if not completely, most systems except those complying strictly with all the requirements in item 2.1.1.11. The mandatory CPL operator used is **C** (strict conjunction) so if any of the items is zero (not present) then the whole structure will evaluate to zero regardless of the other item’s values. In addition, the relative importance of item 2.1.1.11.2 can be seen on the weight assigned to it, which is greater than that assigned to item 2.1.1.11.1.

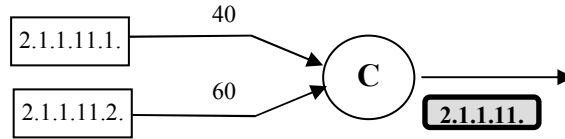


Figure 3. Aggregation Structure for item 2.1.1.11. “Updating of project costs based on”

The aggregation structure for the preference 2.1.3. “Project Costing Adjustments” is shown in Figure 4. In this case, we have assigned different weights in order to reflect the respective importance given to each item.

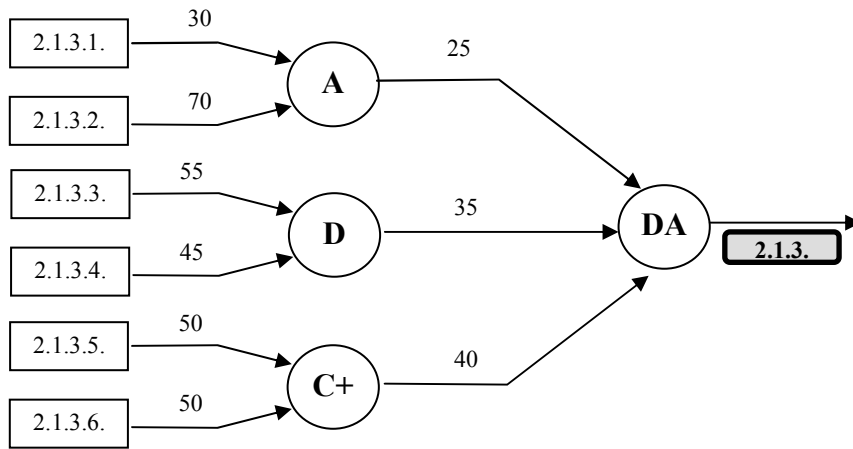


Figure 4. Aggregation Structure for item 2.1.3. “Project Costing Adjustments”

Items 2.1.3.5 and 2.1.3.6 are considered very important and that is why they are aggregated using the mandatory CPL operator **C+**. This operator will return zero if any of the preferences 2.1.3.5 or 2.1.3.6 is missing (zero). We have employed the operator **DA** when aggregating the whole preferences 2.1.3. The CPL operator **DA** is not mandatory, that is the absence of one input does not drop the result to zero, it just penalize it slightly. Preferences 2.1.3.3 and 2.1.3.4. have been aggregated by the operator **D** that returns the maximum value of the aggregated preferences. Finally, we have used the arithmetic media **A** to aggregate preferences 2.1.3.1. and 2.1.3.2..

Figure 5 shows the aggregation structure we have built for item 2.1. “Project Costing”. In this case, we have aggregated items 2.1.1, 2.1.2 and 2.1.3 using a particular structure called *partial absorption* (circled with a dotted line in Figure 5).

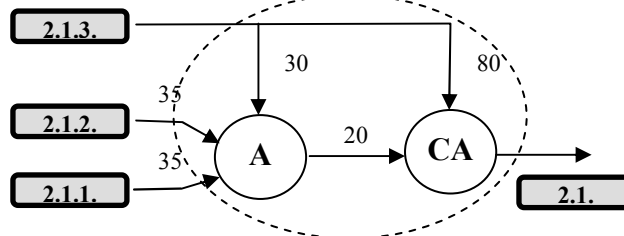


Figure 5. Another example of the Aggregation Structure for item 2.1. “Project Costing”.

Partial absorptions are useful when trying to join a mandatory preference m with a number of optional preferences. If the mandatory preference m is equal to zero then the result of the partial absorption is zero whatever the value of the optional preferences are. Otherwise, the output is the mean of the range $(m-\delta^-, m+\delta^+)$. δ^- y δ^+ determining the weights to be used. δ^- y δ^+ are obtained from a pre calculated table; see [4] for more on this. In this case, preference 2.1.3 has been considered mandatory while preferences 2.1.1 and 2.1.2 have been considered optional.

We have shown through the presented partial aggregation structures the power of the LSP method. It is the responsibility of the analyst to choose the operators and to decide which requirements are mandatory, which are not and how strong the mandatory parts are.

5. Conclusions and Future Work

Nowadays that more and more ERP systems are starting to be used globally, an important asset is having tools to compare and evaluate their different capabilities (scope, characteristics, security, accuracy, etc.). We have presented here one possible model for the evaluation of these systems.

The presented model has been developed following the activities proposed by the LSP method and according to the needs of a target user. Nevertheless, the flexibility of the method employed makes evident the possibility of calibrating the model to new requirements.

Although different models can be tailored to the specific needs of a particular organization, another contribution of the work is in the identification of a number of features common to ERP systems and useful when assessing one of these systems.

Part of our present and future work is focused on using our model considering some of the proposals made in other documents and works.

It is important to remark that cost is an important aspect to be considered when implementing a system, however its evaluation is very complex and it warrants an extensive model, since the different facets to be considered are numerous and not trivial. Items to be considered in this case are, not only equipment's cost, but also maintenance, amortization, storage, transportation, etc. This is another area that is being approached and where we expect to have results to show in future publications.

References

- [1] Chun-Chin Wei, Chen-Fu Chien, Mao-Jiun J. Wang; "An AHP-based approach to ERP system selection". International Journal of Production Economics, Volume 96, Issue 1, 47–62. Elsevier, 2005.
- [2] Constantin Daniel Avram, Razvan Daniel Zota, Laurentiu Ciovisa; "Developing an ERP Strategy Based On the IT Solution Life Cycle". Economy Informatics vol. 12, no. 1/2012 INFOREC Publishing House.
- [3] Dujmovic, J. J. and A. Bayucan, "Evaluation and Comparison of Windowed environments", Proceedings of the IASTED International Conference Software Engineering (SE'97), pp 102-105, 1997.

- [4] Dujmovic, J. J. and Elnicki, R., "A DMS Cost/Benefit Decision Model: Mathematical Models for Data management System Evaluation, Comparison, and Selection", National Bureau of Standards, Washington, D.C., No. NBS-GCR-82-374, NTIS No. PB82-170150 (155 pages), 1982.
- [5] Dujmovic, J. J., "A Method for Evaluation and Selection of Complex Hardware and Software Systems", The 22nd International Conference for the Resource Management and Performance Evaluation of Enterprise Computing Systems. CMG96 Proceedings, vol. 1, pp.368-378, 1996.
- [6] Dujmovic, J. J., "Preferential Neural Networks. Chapter 7 in Neural Networks – Concepts, Applications, and Implementations", Vol. II. Edited by P. Antognetti and V. Milutinovic. Prentice Hall Advanced Reference series, Prentice – Hall, pp 155-206, 1991.
- [7] Dujmovic, J. J., "Quantitative Evaluation of Software", Proceedings of the IASTED International Conference on Software Engineering, edited by M.H. Hamza, pp. 3-7, IASTED/Acta Press, 1997.
- [8] Dujmovic, Jozo J., "Continuous Preference Logic for System Evaluation", IEEE Transactions on Fuzzy Systems, Vol. 15, N° 6, December 2007.
- [9] enVista. www.envistacorp.com. Retrieved April 2013.
- [10] Escalle, Cedric X.; CotteJee, Mark; Austin, Robert D. "Enterprise resource planning (ERP)". Harvard Business School. 1999.
- [11] Infor. <http://www.infor.com>. Retrieved April 2013.
- [12] Jafar Razmi and Mohamad Sadegh Sangari; "A hybrid multi-criteria decision making model for ERP system selection" 4th International Conference on Information and Automation for Sustainability, Sri Lanka, (ICIAfS 2008), IEEE.
- [13] Maya Daneva. "ERP Requirements Engineering Practice: Lessons Learned". IEEE SOFTWARE. March / April 2004
- [14] PROMETHEE, <http://www.promethee-gaia.net/index.html>. Retrieved June 2013.
- [15] Semih Onut and Tugba Efendigil; "A theoretical model design for ERP software selection process under the constraints of cost and quality: A fuzzy approach". Journal of Intelligent & Fuzzy Systems 21 (2010) 365–378. IOS Press.
- [16] simparel. www.simparel.com. Retrieved April 2013.
- [17] SpecIT, <http://www.specit.com>. Retrieved May 2013.
- [18] Stanley Y. W. Su, Jozo Dujmovic, D. S. Batory, S. B. Navathe, and Richard Elnicki. "A Cost-Benefit Decision Model: Analysis, Comparison, and Selection of Data Management Systems". ACM Transactions on Database Systems, Vol. 12, No. 3, September 1987.
- [19] T. C. Wang, Y. C. Chiang, S. C. Hsu "Applying Incomplete Linguistic Preference Relations to a Selection of ERP System Suppliers". Proceedings of the Industrial Engineering and Engineering Management, IEEE International Conference IEEM 2007.
- [20] TEC Technology Evaluation Centers. <http://www.technologyevaluation.com>. Retrieved April 2013.
- [21] Ufuk Cebeci; "Fuzzy AHP-based decision support system for selecting ERP systems in textile industry by using balanced scorecard". Expert Systems with Applications 36 (2009). ELSEVIER.
- [22] W.-H. Tsai, P.-L. Lee1, Y.-S. Shen, C.-C. Yang. "The relationship between ERP software selection criteria and ERP success". Proceedings of the 2009 IEEE IEEM.
- [23] Xavier Franch and Juan Pablo Carvallo "Using Quality Models in Software Package Selection"- IEEE SOFTWARE. January / February 2003. IEEE Computer Society.
- [24] Ya-Yueh Shih; "A study of ERP systems selection via fuzzy AHP method". Proceedings of the Information Engineering and Electronic Commerce, 2nd International Symposium 2010, (IEEC 2010). IEEE.