Reengineering the Knowledge Component of a Data Warehouse-Based Expert Diagnosis System

Jean-François Beaudoin[1], Sylvain Delisle[1], Mathieu Dugré[1], and Josée St-Pierre[2]

[1] Département de mathématiques et d'informatique
[2] Département des sciences de la gestion
Institut de recherche sur les PME,
Laboratoire de recherche sur la performance des enterprises,
Université du Québec à Trois Rivières,
C.P. 500, Trois-Rivières, Québec, Canada, G9A 5H7
Phone: 1-819-376-5011 + 3832
Fax: 1-819-376-5185
{jean-francois_beaudoin, sylvain_delisle, mathieu_dugre,
josee_st-pierre}@uqtr.ca
www.uqtr.ca/{~delisle, dsge}

Abstract. We describe the weaknesses of an existing expert diagnosis-recommendation system we have developed for SMEs. In good part, these weaknesses are related to the fact that the system was not implemented with appropriate artificial intelligence techniques. We recently decided to tackle the problem and re-engineered the core of the system with the help of an up-to-date expert system shell. In the process, we revised the formalization and reorganization of the system's expertise and developed a brand new knowledge base. We here describe the new system and the improvements made, and we identify ongoing and future developments.

1 Introduction

In 1999, we developed an expert diagnosis system for small and medium-sized enterprises (**SME**s), the PDG system [1, 2]. This system is based on a benchmarking approach [3-5] and performs a multidimensional evaluation of a SME's production and management activities, and assesses the results of these activities in terms of productivity, profitability, vulnerability and efficiency. This system is fully operational and has been put to use on actual data from more than 500 SMEs from Canada, USA, and France. By academic standards, it is clearly a successful real-life application [2].

What is peculiar though, especially from a knowledge-based systems perspective, is the fact that although the PDG system is packed with knowledge and expertise on SMEs, it has not been originally implemented with "traditional" symbolic Artificial Intelligence (**AI**) techniques due to lack of time, human and financial resources. Today, we must admit that this implementation decision was not optimal as it is the main cause for certain weaknesses in the system's knowledge component, including its lack of flexibility, difficulty in understanding, and limited capacity for adaptation, improvement and updating.

K.V. Andersen, J. Debenham, and R. Wagner (Eds.): DEXA 2005, LNCS 3588, pp. 910 – 919, 2005.

We are currently working on the development of a second generation PDG system, which we call the PDGII system. Although it is based on the first system, two key components have undergone in-depth reengineering: its database, which has now become a powerful data warehouse, and its knowledge base and reasoning engine, which are being re-designed and re-implemented with symbolic AI techniques [6]. In what follows, we first explain the reasons justifying the development of the new PDGII system and we identify the goals we have set ourselves in this second generation system. Then, the main part of the paper consists in a discussion of the new knowledge-based subsystem: the selected AI techniques and tools, the formalization of the expertise, and the additions and benefits brought along. We also briefly talk about potential future developments in the PDGII system.

Our work takes place within the context of the Research Institute for SMEs. The Institute's core mission is to support fundamental and applied research to foster the advancement of knowledge on SMEs to contribute to their development. Our lab, the LaRePE *(LAboratoire de REcherche sur la Performance des Entreprises*: www.uqtr.ca/ inrpme/ larepe/), is mainly concerned with the development of scientific expertise on the study and modeling of SMEs' performance, including a variety of interrelated subjects such as finance, management, information systems, production, technology, etc. All research projects carried out at the LaRePE involve both theoretical and practical aspects, always attempting to provide practical solutions to real problems confronting SMEs, often necessitating in-field studies.

2 Towards the Reengineered Diagnosis System

The initial PDG system was and still is a good diagnosis system [2]. With the use of a lot of data collected from a comprehensive questionnaire filled by an evaluated SME, this system identifies and evaluates the enterprise's weaknesses. Then, relevant recommendations are suggested to help the evaluated SME correct its weaknesses and thus improve its performance. The questionnaire's data are stored in a database (now a data warehouse) and the SAS statistical package is used to perform various statistical calculations. Thereafter, the PDG system imports these statistical data to perform the diagnosis and to produce a detailed evaluation report in which we find the results presented as graphics and texts. The whole system, from the diagnosis program to the report production program, was originally developed with Microsoft Excel. This system is fully functional since 1999 and still produces performance diagnostic reports very much appreciated by SME owners-managers.

However, despite its success and correctness, the original PDG system was out of date with the current state of the art in this kind of system [7-9] and suffered from important gaps and weaknesses that made its functioning, updating, understanding, and evolution rather difficult. The main element of any expert system is a knowledge base in which resides the system's expertise. The fact is that the original PDG system does not have an explicit knowledge base. Indeed, the expertise elements are scattered throughout the Excel programming code and cells. Consequently, it is extremely difficult to find and update expertise elements, even more to ensure that changes will not result in the introduction of unforeseen consequences. The usability and flexibility of the code is reduced dramatically by this weakness. The high coupling between the code and the expertise makes the improvement of the system excessively difficult: this is definitely a major handicap for a good expert system that needs to evolve with our grasp of the application domain, especially in the long run.

Moreover, although the final diagnosis reports produced by the PDG system are of the highest quality, the report production component is not flexible at all. The code associated with the production of the report is intermingled with the code supporting the diagnosis system's expertise. Thus, it is impossible to modify the report without affecting the rest of the PDG system. For example, simply removing, adding, or modifying a graphic in the report involves a considerable programming effort. So, there is a harmful coupling between the diagnosis expert system and the report generation elements, especially those dealing with graphical representations.

Our laboratory is currently working on a brand new PDGII system that will correct these weaknesses. This new SME performance diagnosis system is a complete reengineering of the old system and is built on a solid artificial intelligence basis. Here are the goals we have set ourselves with regard to the knowledge-base component:

- Centralize and formalize all the expertise elements in a flexible and well-structured knowledge base.
- Replace Excel by another programming tool more adapted to our situation and needs.
- Separate the programming code of the diagnosis system from the programming code of the report's graphical representations.
- Encapsulate in a flexible way the various objects involved in a diagnosis so that they will be able to easily support new reports and specific user profiles.
- Revise the graphics production mechanism to support in a flexible way more personalized final reports.
- Devise a structure and mechanism that will save in our data warehouse all the detailed diagnosis results.

3 The New Knowledge-Based Component

3.1 Selecting the Appropriate AI Tool

To implement the new PDGII system, we had to find appropriate, new programming tools. We needed to program the newly reengineered expert system, based on up-to-date AI techniques [6], plus other more conventional (non-AI) elements. As to the main conventional implementation language, we chose Java. Since most of the systems in our laboratory are now programmed in Java, practical considerations motivated this choice. As far as the knowledge-based component is concerned, we needed a capable and efficient tool that would support the creation of a knowledge base and the use of an inference engine. In order to find the best AI tool to suit our needs, we conducted a quite extensive comparative evaluation based on information available on vendor's Web sites and also on several applied AI research papers. We even tested some of these tools on our systems. In the end, the winner was Flex.

Flex [10] is a tool from Logic Programming Associates Ltd (www.lpa.co.uk) and is especially designed for the development of expert systems, including both the knowledge base and the inference engine components. Flex knowledge bases are "frame-based” [11] and they are easy to develop. Flex also has its own inference engine. The knowledge base and the inference rules must be written in the Flex KSL (Knowledge Specification Language) [10]. An important feature of the Flex KSL language is that it is close to English in many ways, so it is easy to use and to understand. Here is an example that illustrates how easy it is to define a frame and an instance of the latter:

```
frame  evaluation_criterion ;
       default  description  is  ‘Complete the description for the criterion.’

frame  benchmark_data ;
       default  enterprise_value          is  _  and
       default  reference_group_value     is  _  and
       default  benchmark_value.

frame  technological_proficiency  is a  evaluation_criterion ;
       default  data1  is a  benchmark_data  and
       default  data2  is a  benchmark_data  and
       default  data3  is a  benchmark_data  and
       default  data4  is a  benchmark_data  .

instance  criterion1  is a  technological_proficiency  .
```

In this example, we can see the use of the inheritance mechanism. We have the **evaluation_criterion** parent frame and its more specialized children frame **technological_proficiency**. The children frame inherits all the attributes of the parent frame, unless otherwise specified. The type of these children attributes is also a frame. So, each attribute of the children frame will be of type **benchmark_data**. Finally, we can see how easy it is to create an instance of the children frame.

We could have specified explicit values for the attributes of the instance **criterion1**, but as it is possible to build Flex procedures in KSL (or even in Prolog), we could also have specified the attributes data by programming, as in:

```
do  criterion1’s  data1’s  enterprise_value  becomes 232.67 .
```

The Flex inference engine supports both forward and backward chaining. Here is an example showing the simplicity with which one can write a forward chaining rule:

```
rule benchmark_value_verification
       if   S  is an instance of   evaluation_criterion   and
              S`s   data1`s enterprise_value  >  S`s  data1`s  reference_group_value
       then  S`s  data1`s  benchmark_value  becomes  ‘Enterprise is better’ .
```

In addition to Flex, LPA also offers other interesting and useful tools for the development of AI-based systems. The Flint tool [12] supports the management of uncertainty in an expert system through these AI techniques: fuzzy logic, Bayesian updating, and certainty factors—this is a tool we will reconsider later in the development of our new PDGII system. With yet another LPA tool, the ProData Interface [13], it is possible to use a database from within a Flex-based system. This is an important capability in the PDG system as it must absolutely be able to get access to the data warehouse containing all SME-related data. Another useful LPA tool is the Intelligence Server [14], which allows an external (or foreign) application to connect to a Flex knowledge base and submit requests to it. Consequently, we can create the non-AI subset of the new PDGII system in Java and work with the Flex knowledge base through the Intelligence Server interface.

3.2 System Architecture

We now consider in more detail how the different tools mentioned in Section 3.1 are actually organized in the PDGII system: see Figure 1 below. First of all, there is a Java program acting as the entry point of the system; this program controls the execution of the entire system. A diagnosis editor allows the user to enter essential information and parameters about the configuration of the diagnosis to be produced, such as the evaluated SME's identification, the information for the creation of the reference group (against which the evaluated SME will be compared), the desired diagnosis type, etc. Another important element for the PDGII system is that it is possible, during this configuration phase, to select and modify some expertise elements (via the Intelligence Server interface), such as weights, that will be used for the performance diagnosis. For testing purposes, such flexibility in this configuration phase is quite useful when we want to measure the impact of some parameters on the system's behaviour without making any other changes.

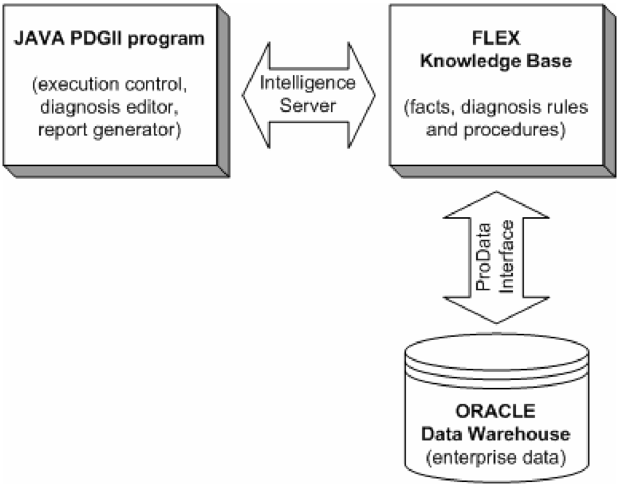

Fig. 1. The PDGII system architecture

It is also possible to select the desired profile for a PDGII diagnosis. We will come back to this later, but let us just say that several different profiles of diagnosis can be created in order to support the evaluation of SMEs from different perspectives. In fact, the knowledge base contains predetermined profiles of PDGII diagnoses. The user simply has to select the one he/she desires. For example, the complete profile consists in the performance evaluation of all activities of an SME. Yet another profile consists in the evaluation of, only, the production and management systems.

Once the diagnosis has been launched, the knowledge base has to communicate with the Oracle data warehouse to download all the relevant SME data. Our data warehouse has been built one year ago [15] and is still the subject of ongoing work. This new data warehouse supports the expert diagnosis system, and is also used in various other research projects on SME data. Communication between the Flex knowledge base and the data warehouse is made possible through the ProData interface. The Flex knowledge base then performs the diagnosis with the help of its facts, rules and procedures. At the end of the diagnosis, the results are first uploaded in the Oracle database (data warehouse) and then used by the report maker from the Java program. At this stage, a report configuration can be specified by the user.

4 Expertise Organization and Formalization

4.1 The Knowledge Base Structure

In the initial version of the PDG system, expertise was scattered throughout the Excel implementation. It was extremely difficult to locate and understand expertise elements, and ensure their safe maintenance. Because of the crucial role played by the PDG system's knowledge component, the finding of a solution to this problem very much influenced the design of the new PDGII system. The initial phase of the reengineering was thus the identification (and understanding) of all the expertise elements dispersed in the original Excel implementation—a difficult and tedious task since the system had been programmed over a period of several years, by different programmers, and supported by essentially no documentation. Then, all expertise elements were verified, centralized, and organized into a well-structured, frame-based Flex knowledge base.

We also wanted this knowledge base to be usable by other systems, not only the PDGII system. So it had to be devised in a relatively generic way. Indeed, throughout the years, our laboratory has built a strong expertise in SME performance evaluation. This expertise has translated into the development of performance evaluation criteria in several key SME-related domains. In fact, an expert system such as the PDGII system would not be possible without these precious criteria belonging to various domains. Thus, we decided to build a knowledge base that could be used by any diagnosis system in our laboratory, not only the PDGII system. A high degree of flexibility and reusability was a goal of the utmost importance in this phase of our work.

Because these SME evaluation criteria are not necessarily specific to the PDGII system, it was important to make this distinction in the knowledge base. Consequently, we organized the knowledge base in two parts. The first part (left-hand side in Figure 2) contains all the expertise that can be used by any of our diagnosis systems, i.e. which is generic and not PDGII-specific. This subset of the knowledge base contains 64 evaluation criteria. The second part (right-hand side in Figure 2) of the knowledge base contains all the expertise elements that are specific to each diagnosis system, such as the PDGII system. Figure 2 illustrates how the knowledge base is organised.

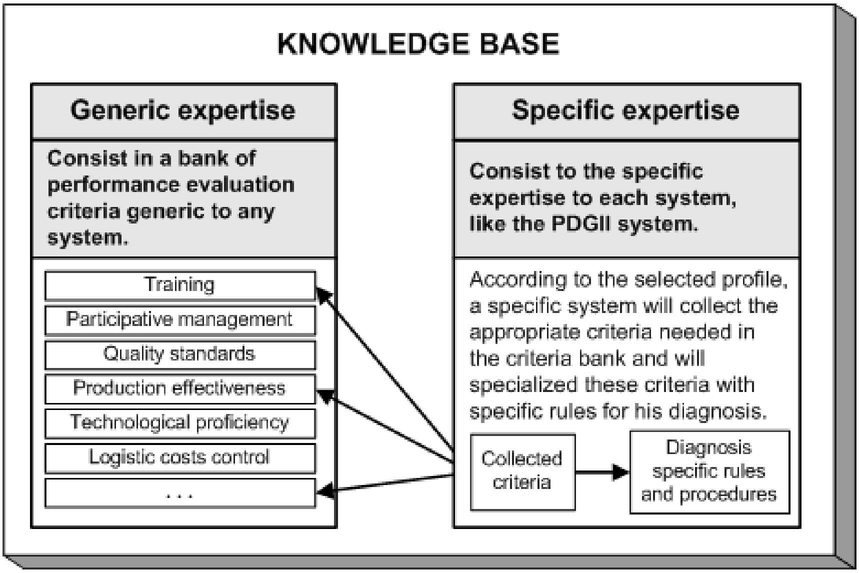
KNOWLEDGE BASE
Generic expertise
Consist in a bank of performance evaluation criteria generic to any system.
Training
Participative management
Quality standards
Production effectiveness
Technological proficiency
Logistic costs control
. . .
Specific expertise
Consist to the specific expertise to each system, like the PDGII system.
According to the selected profile, a specific system will collect the appropriate criteria needed in the criteria bank and will specialized these criteria with specific rules for his diagnosis.
Collected criteria
Diagnosis specific rules and procedures

Fig. 2. The expertise general to every diagnosis system appears on the left-hand side, and the expertise specific to each diagnosis system on the right-hand side

Another important concept in the organisation of the knowledge base is the link between the generic and the specific parts. As mentioned before, the PDGII system uses different diagnosis profiles, each defining a specific orientation of the diagnosis performed by the PDGII system. For example, one specific PDGII profile could define a specialized diagnosis of the SME's human resources activities, while another PDGII profile could define the general diagnosis of all activities. Each profile is defined by the evaluation criteria needed to perform the associated diagnosis. Indeed, each profile selects the required criteria in the generic subset of the knowledge base, and then complements these criteria with PDGII-specific knowledge (e.g. PDGII data, PDGII rules, etc.) to allow the computation of the diagnosis in the specific context of the PDGII system.

4.2 Levels of Formalization in the PDGII Expertise

Another major improvement implemented in the PDGII system's knowledge base is the three-level expertise formalization that was carried out. To better understand these levels, let us first take a look at the structure of the PDGII system's diagnosis. The latter performs a performance diagnosis of an evaluated SME in terms of results and management practices in different activity sectors (and business functions), relative to a reference group of similar SMEs. Each activity sector relies on evaluation criteria for a sector-specific diagnosis—these criteria are organized appropriately in the knowledge base as explained in Section 4.1. Figure 3 below presents the three-level PDGII diagnosis structure and the three-level expertise associated with them.

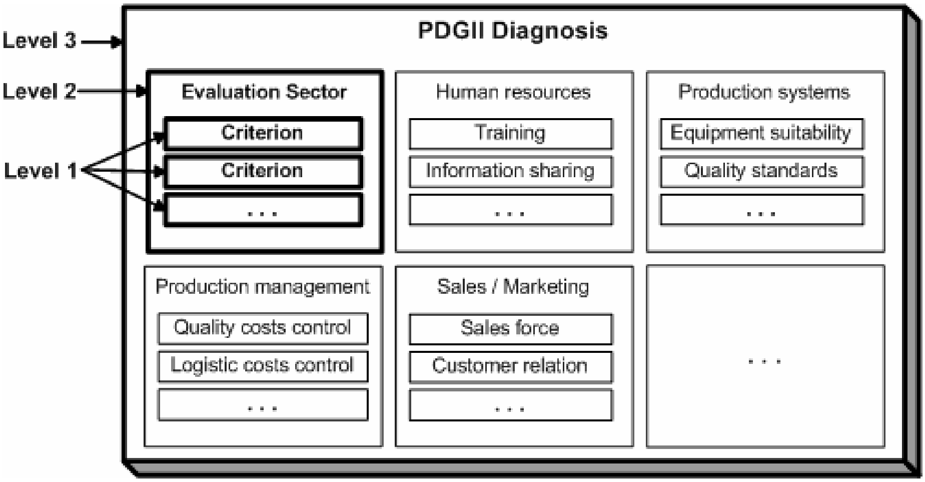

Fig. 3. The PDGII diagnosis system structure and its three-level expertise

The first level corresponds to all the generic and specific evaluation criteria that the PDGII system has selected in the knowledge base, as explained in Section 4.1. These expertise elements include data related to the evaluated SME and the SMEs in the reference group (for benchmarking purposes), and the PDGII-specific rules needed by the relevant criteria. Moreover, this first expertise level also contains all the rules allowing the production of the comments and recommendations on the evaluated SME's performance on the selected criteria. So, all the expertise elements used during the individual evaluation of the criteria of each selected activity sector, according to the active profile, are part of this first expertise level.

The knowledge base's second expertise level corresponds to the expertise on the global evaluation of each activity sector (or business function) of a PDGII diagnosis. Each evaluation criteria of each sector are grouped together to perform this global sector evaluation. Playing a central role among the elements of this second expertise level is the weight of each evaluation criterion relative to its sector. We also find all the rules allowing the global performance diagnosis of each activity sector, and also the rules allowing the identification of the weakest criteria of each sector for the evaluated SME. Once the identification of the weakest criteria of each sector is done, comments and recommendations associated with these criteria are produced with the help of the first expertise level.

The third expertise level of the knowledge base is associated with the global evaluation of all activity sectors of a PDGII diagnosis profile. It corresponds to a complex and sophisticated level of expertise because we must be able to compare and balance different sectors of activity within the evaluated SME, while being able to explain why it does better or worse than its reference group and if this needs attention or action in the near future. Thus, it still is a challenging task for SME performance evaluation experts to find reliable expertise rules to model this kind of global evaluation. At present time, a human expert always revises the evaluation automatically produced by the PDG system at this level. Thus, this third expertise level is not totally implemented yet in the current version of the knowledge base. And this is part of our future developments in the PDGII system.

5 Conclusion

In this paper, we presented the knowledge-based component of our new expert system on SME performance diagnosis, the PDGII system. Although the initial version of the PDG system managed to produce SME evaluations and reports of the highest quality, the implementation rendered maintenance and evolution of the system a daunting task. In particular, the fact that the expert system did not use available AI techniques and tools, and the fact that expertise elements were scattered throughout the Excel code and cells, made any evolution of the system a risky business. Another major problem was the tight coupling between the evaluation expertise and the report generation details in the implementation. Thus, even the smallest modification made to the report produced by the diagnosis system was a great challenge and often led to unforeseen consequences. For all these reasons, and despite the success of the initial version, we made the decision the move along and reengineer the PDG system, leading to the new PDGII system presented here.

Several important improvements were made along the way. The choice of LPA's Flex software for the knowledge-based component of the PDGII system, as well as the use of the Java programming language contributed to adequate integration at the software level, especially in the context of the software architecture of our laboratory. The two main components in our new PDGII system are the data warehouse and the knowledge-based expert system. An important task of expertise organization and formalization was carried out, as explained in Section 4, to regroup the PDG's diagnosis expertise under three different but complementary levels. Moreover, this knowledge base was developed with the goal of explicitly distinguishing generic knowledge (i.e. used in several diagnosis systems) from specific knowledge (i.e. used in only one specific diagnosis system).

Another advantage of the new PDGII system is that the user has a better control over the diagnosis. Indeed, a user has the possibility to configure some expertise elements of the knowledge base during the configuration phase of a diagnosis. The PDGII system has also gained in control and flexibility with the inception of diagnosis profiles that will affect the diagnosis on different activity sectors of the evaluated SME. Profiles also allow the user to obtain personalized evaluation reports. These improvements represent significant benefits both to the end users, i.e. owners-managers, and the PDG development team.

We are currently performing an extensive validation in which we produce, in parallel, the performance diagnosis with both the new PDGII system and the existing PDG system, from the same data. We then make a detailed comparison of the results in order to identify potential bugs in the new system. So far, more than ten full diagnosis comparisons have been made and only minor bugs have been uncovered.

As it has been done for the last ten years, our laboratory continues to develop and formalize its expertise in SME performance evaluation. The new PDGII system makes a significant contribution in that regard. As to future work, the performance diagnosis results saved in our data warehouse will be used for data mining. Also, another important future work item will be the completion of the third level expertise of our knowledge base as presented above.

Acknowledgements

The authors acknowledge the Natural Sciences and Engineering Research Council of Canada and the J.-Armand Bombardier Foundation for their financial support.

References

1. St-Pierre, J. and S. Delisle, *An Expert Diagnosis System for the Benchmarking of SMEs' Performance.* Benchmarking-An International Journal, to appear.
2. Delisle, S. and J. St-Pierre, *Expertise in a Hybrid Diagnostic-Recommendation System for SMEs: A Successful Real-Life Application.* 3029, Lecture Notes in Computer Science, Spinger-Verlag. 2004. pp.807-816. (Winner of the best paper award)
3. St-Pierre, J., L. Raymond, and E. Andriambeloson. *Performance Effects of the Adoption of Benchmarking and Best Practices in Manufacturing SMEs.* in *Conference on Small Business and Enterprise Development*. 2002. The University of Nottingham (UK).
4. Cassell, C., S. Nadin, and M.O. Gray, *The Use and Effectiveness of Benchmarking in SMEs.* Benchmarking: An International Journal, 2001. **8**(3): pp.212-222.
5. Yasin, M.M., *The Theory and Practice of Benchmarking: Then and Now.* Benchmarking: An International Journal, 2002. **9**(3): pp.217-243.
6. Shu-Hsien Liao, *Expert System Methodologies and Applications--A Decade Review from 1995 to 2004.* Expert Systems with Applications, 2005. **28**(1): pp.93-103.
7. Carlsson, C. and E. Turban, *DSS: Directions for the Next Decade.* Decision Support Systems, 2002. **33**(2): pp.105-110.
8. Nedovic, L. and V. Devedzic, *Expert Systems in Finance--a Cross-section of the Field.* Expert Systems with Applications, 2002. **23**(1): pp.49-66.
9. Shim, J.P., et al., *Past, Present, and Future of Decision Support Technology.* Decision Support Systems, 2002. **33**(2): pp.111-126.
10. LPA, *Flex Expert System Toolkit: Flex Reference.* 1996, London: Logic Programming Associates Ltd.
11. Turban, E. and J.E. Aronson, *Decision Support Systems and Intelligent Systems.* 2001: Prentice Hall.
12. LPA, *Flint Reference*. 2004, London: Logic Programming Associates Ltd.
13. PA, *ProData Interface*. 2004, London: Logic Programming Associates Ltd.
14. LPA, *Intelligence Server*. 2004, London: Logic Programming Associates Ltd.
15. Delisle, S., M. Dugré, and J. St-Pierre. *Multidimensional SME Performance Evaluation: Upgrading to Data Warehousing & Data Mining Techniques. Proceedings of the International Conference on Information and Knowledge Engineering*. 2004. Las Vegas, Nevada: CSREA Press.