

2012

Using metarules to integrate knowledge in knowledge based systems. An application in the woodworking industry

Alvaro Villavicencio
University of Northern Iowa

Copyright ©2012 Alvaro Villavicencio

Follow this and additional works at: <https://scholarworks.uni.edu/etd>

 Part of the [Industrial Engineering Commons](#)

Let us know how access to this document benefits you

Recommended Citation

Villavicencio, Alvaro, "Using metarules to integrate knowledge in knowledge based systems. An application in the woodworking industry" (2012). *Electronic Theses and Dissertations*. 613.
<https://scholarworks.uni.edu/etd/613>

This Open Access Dissertation is brought to you for free and open access by the Graduate College at UNI ScholarWorks. It has been accepted for inclusion in Electronic Theses and Dissertations by an authorized administrator of UNI ScholarWorks. For more information, please contact scholarworks@uni.edu.

USING METARULES TO INTEGRATE KNOWLEDGE
IN KNOWLEDGE BASED SYSTEMS.
AN APPLICATION IN THE WOODWORKING INDUSTRY

A Dissertation

Submitted

In Partial Fulfillment

of the Requirements for the Degree

Doctor of Industrial Technology

Approved:

Dr. MD Salim, Chair

Dr. John T. Fecik, Co-Chair

Dr. Nageswara Rao Posinasetti, Committee Member

Dr. Ali E. Kashef, Committee Member

Dr. John W. McCormick, Committee Member

Dr. Andrew R. Gilpin, Committee Member

Alvaro Villavicencio

University of Northern Iowa

May 2012

UMI Number: 3528374

All rights reserved

INFORMATION TO ALL USERS

The quality of this reproduction is dependent upon the quality of the copy submitted.

In the unlikely event that the author did not send a complete manuscript and there are missing pages, these will be noted. Also, if material had to be removed, a note will indicate the deletion.

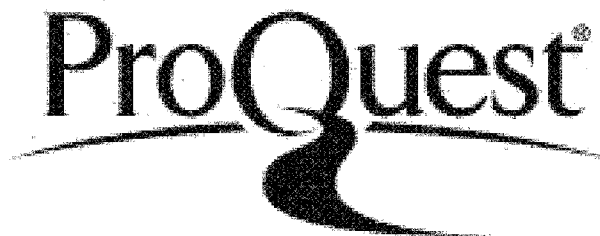


UMI 3528374

Published by ProQuest LLC 2012. Copyright in the Dissertation held by the Author.

Microform Edition © ProQuest LLC.

All rights reserved. This work is protected against unauthorized copying under Title 17, United States Code.



ProQuest LLC
789 East Eisenhower Parkway
P.O. Box 1346
Ann Arbor, MI 48106-1346

Copyright by

Alvaro Villavicencio

2012

All Rights Reserved

USING METARULES TO INTEGRATE KNOWLEDGE
IN KNOWLEDGE BASED SYSTEMS.
AN APPLICATION IN THE WOODWORKING INDUSTRY

An Abstract of a Dissertation

Submitted

In Partial Fulfillment

of the Requirements for the Degree

Doctor of Industrial Technology

Approved:

Dr. MD Salim, Committee Chair

Dr. Michael Licari
Dean of the Graduate College

Alvaro Villavicencio
University of Northern Iowa

May 2012

ABSTRACT

The current study addresses the integration of knowledge obtained from Data Mining structures and models into existing Knowledge Based solutions. It presents a technique adapted from commonKADS and spiral methodology to develop an initial knowledge solution using a traditional approach for requirement analysis, knowledge acquisition, and implementation. After an initial prototype is created and verified, the solution is enhanced incorporating new knowledge obtained from Online Analytical Processing, specifically from Data Mining models and structures using meta rules. Every meta rule is also verified prior to being included in the selection and translation of rules into the Expert System notation. Once an initial iteration was completed, responses from test cases were compared using an agreement index and kappa index.

The problem domain was restricted to remake and rework operations in a cabinet making company. For Data Mining models, 8,674 cases of Price of Non Conformance (PONC) were used for a period of time of 3 months.

Initial results indicated that the technique presented sufficient formalism to be used in the development of new systems, using Trillium scale. The use of 50 additional cases randomly selected from different departments indicated that responses from the original system and the solution that incorporated new knowledge from Data Mining differed significantly. Further inspection of responses indicated that the new solution with additional 68 rules was able to answer, although with an incorrect alternative in 28 additional cases that the initial solution was not able to provide a conclusion.

ACKNOWLEDGEMENTS

I would like to express my deepest gratitude to all the faculty, friends and family members that were involved in this work in many ways. Reaching to an end in my degree would have been impossible without their guidance, patience, advice and support through this long and challenging journey. Nonetheless, I express my deepest gratefulness to the following people that contributed to my advanced education.

First, a special thank you to my advising committee, Drs. Salim, Fecik, Kashef, Rao, Gilpin and McCormick for their support, assistance, help and counseling in challenging moments. My special gratitude goes to my Committee Chair, Dr. MD Salim, and Committee Co-Chair, Dr. John T. Fecik for their special dedication, caring, concern, and extra time that helped me to complete this dissertation.

I also like to express my sincere appreciation to Dr. Recayi Pecen, Dr. Mohammed F. Fahmy, and Dr. James Maxwell, and the Graduate Committee for helping me and encourage me to finish my degree, especially on my final stages.

To my colleges and friends that helped me in the academic discussion of many topics.

Finally, a very special thank you to my parents Geldy and Carlos, and Grandmother Veronica for inspire me with values and virtues that allow me to reach my dreams. A special mention and thank you to my wife, Francys for her patience, dedication, support, and help that encourage me to complete my dissertation, and to my son Francisco and my daughters Maureen and Carolina for their sacrifice in this adventure.

TABLE OF CONTENTS

LIST OF TABLES	v
LIST OF FIGURES	vi
CHAPTER I. INTRODUCTION.....	1
Statement of the Problem.....	7
Statement of the Purpose	8
Statement of Need.....	9
Statement of Research Questions.....	14
Assumptions.....	15
Delimitations.....	15
Definition of Terms.....	16
CHAPTER II. REVIEW OF RELATED LITERATURE	29
Knowledge Based System Development.....	29
Meta-knowledge Techniques and Knowledge Management in KBS	37
Artificial Intelligence in OLAP, Data Warehouse, and Data Mining.....	42
CHAPTER III. METHODOLOGY	46
Research Design.....	48
Variables and Instruments.....	51
Experimental Procedures and Data Analysis.....	53

CHAPTER IV. RESULTS AND ANALYSIS OF DATA.....	58
Summary of Procedures.....	58
Step 1. Development of a Knowledge Based System using commonKADS.....	58
Step 2. Data Warehouse implementation for EOL PONC information.....	70
Step 3. Meta rule System Development for analysis of rules obtained from Analysis Services	81
CHAPTER V. SUMMARY, CONCLUSIONS AND RECOMMENDATIONS	85
Report of Results.....	85
Conclusions.....	95
Recommendations.....	97
REFERENCES	99
APPENDIX A. PONC EXPERT SOURCE CODE	106
APPENDIX B. METARULES SOURCE CODE	127
APPENDIX C. MODEL VIEWER AND RULE EXTRACT SOURCE CODE.....	137
APPENDIX D. DMMRULES FILE CONTENT	149
APPENDIX E. NEWRULES FILE CONTENT	153
APPENDIX F. 50 CASES AND PONC EXPERT RESPONSE.....	161

LIST OF TABLES

1. Trillium Scale description.....	49
2. Structural deficiencies of Rule Bases	50
3. Logic Operand equivalents in RO-RA-RV form	51
4. Redundant or conflicting Rule Cases.....	54
5. Presentation of results and classification of agreement for each variable	55
6. Rules obtained after knowledge refinement	64
7. Trillium scale results for proposed technique.....	87
8. Meta rules used to select and map rules for PONC Expert, expressed in IF (LHS) THEN (RHS) form.....	88
9. Rule antecedent and consequent in standard RO-RA-RV form for meta rules	89
10. PONC Expert pair comparison results.....	93

LIST OF FIGURES

1. Traditional Decision System Development Cycle	4
2. New decision paradigm in Decision Making System development	5
3. Data Warehouse components.....	17
4. Relationship between Data Warehouse, OLAP and Data Mining on current DSS paradigm	18
5. Expert System Development Cycle	21
6. KADS methodology stages.....	23
7. Simple model of Expert System components	24
8. OLAP cube representation.....	26
9. CommonKADS model.....	32
10. Protégé general usage	34
11. Knowledge management cycle	39
12. Semantic Scale for agreement classification of solutions.....	52
13. Spiral Development approach.....	59
14. CommonKADS methodology models	60
15. PONC related objects and attributes used in knowledge identification.....	63
16. Project PONC Expert code structure	65
17. General PONC Expert interface.....	66
18. PONC Expert with a true/false question object in answer window.....	67
19. PONC Expert Results screen region with complete session interaction.....	68
20. RuleSet obtained through knowledge refinement.....	69
21. Dimensions and measurements defined for EOL PONC data warehouse.....	71

22. Data source view model for EOL PONC data warehouse cube definition and data mining	72
23. One dimension and one measurement for multidimensional analysis (cubes).....	73
24. Multidimensional cube with dimension projection for EOL PONC	74
25. Learning machine view.....	75
26. Data Mining models in Analysis Services 2005 for department and value rules	76
27. Clementine model for PONC department estimation using association rules.....	77
28. Rule selection for item sets defined in Analysis Services 2005 for EOL PONC in tabular form.....	78
29. Candidate rules table obtained in Clementine with correspondent item sets for EOL PONC cases	78
30. Candidate rules obtained in Clementine	79
31. VB .NET 2005 program used to save obtained rules from Association rules method in a file for further processing.....	80
32. Data mining candidate rules exported to file	81
33. Meta rules embedded in automated program used to create final rules to be used in PONC Expert	82
34. Metarules processing screen	83
35. Rules obtained after meta rules analysis in a standard form to be included in PONC Expert System	83
36. Modified technique for PONC Expert development	86
37. Data mining new rules and its inclusion in PONC Expert.....	91

CHAPTER I

INTRODUCTION

Artificial Intelligence (AI) created multiple expectations and delivered mixed results among the scientific community. Since its introduction in the 1950s as a separate discipline, AI nurtured the idea of intellectually complex machines available by the end of the century, as noted by Lenat (2008). Its applications have a vast implication in many areas, and have been addressed by several authors in different disciplines such as medicine, chemistry, design, avionics, education, and mathematics. Regardless of its significant potential contribution in these areas, it has received ambivalent evaluations, as suggested by Tommelein, Levitt and Hayes-Roth (1992), Lenat (2008), and Reich (1996). The authors cited several reasons for this ambivalence, which are grouped in three main factors: (a) inability to cope with initial expectations for the discipline, (b) inadequate and insufficient reports compared to the number of commercial applications, and (c) differences in modeling results and knowledge that they represent.

An initial purpose of the AI field was to build intelligent machines by simulating human behavior, an elusive concept as indicated by Negnevitsky (2005) and Hopgood (2003). In order to obtain an intelligent machine, AI evolved in different areas each of them developing its own technology to achieve this goal as noted by Brackenbury and Ravin (2002) and Bronzino and Morelli (1989). The main areas indicated by the authors comprise: Natural language understanding (NLU), machine reasoning, knowledge representation and acquisition, dialog management, intelligent tutoring and emotion. Each of these areas presents challenges and state of the art techniques for addressing

specific problems in their stable problem domain. Today, Artificial Intelligence exhibits much more ambitious although specific goals, as noted by Anthes (2009) and its applications are embedded on software programs or distributed agents that users access transparently every day.

Among these areas of development, Knowledge Based Systems (KBS) emerged as a successful area of Artificial Intelligence, with a significant impact in the business arena. By producing software components that encapsulate technical expertise, explain their conclusions, and manipulate symbolic reasoning on a specific domain of expertise (Giarratano & Riley, 1998; Jackson, 1999), it appears as a natural intelligent collaboration tool in management processes. As economies switch from information based paradigm into knowledge based ones, new challenges emerge and new tools are required to address its increasing complexity (Harmon & Sawyer, 1990).

This paradigm shift introduced other techniques from databases, such as Online Analytical Processing (OLAP), Data Mining, and Data Warehousing (DW). These techniques represented an important contribution in data integration, advanced reporting, data analysis and multidimensional storage, capturing the attention of the commercial and research communities (Lechtenborger & Vossen, 2003; Ma, Chou, & Yen, 2000; Pedersen, Jensen, & Dyreson, 2001). Although these techniques offer advanced mathematical principles used for automatic knowledge discovery and data analysis as in the case of Data Mining, they do not provide a complete solution by themselves and require integration efforts among them and with other disciplines for a comprehensive solution and integration with existing transactional and decision making processes, as in

the case of OLAP and Data Warehousing (Pendse, 2001). However, Shim et al. (2002) indicated that these three tools (OLAP, Data Warehouse, and Data Mining) are part of future trends in Decision Support Systems (DSS).

It is precisely in the area of decision support where changes occur more rapidly. Since the paradigm moved from data centered decision to knowledge-based, the initial approach of modeling decision processes mainly centered on a transactional approach (Shim et al., 2002; Weitzel & Kershberg, 1989), as presented in Figure 1. Because of its inherent uncertainty of being a numerical and statistical based technique, decision processes face additional difficulties due to increasing complexity on the user side. More sophisticated and demanding requirements are modeled and global organizations become more complex in nature and flexible in their interaction as suggested by Shim et al (2002).

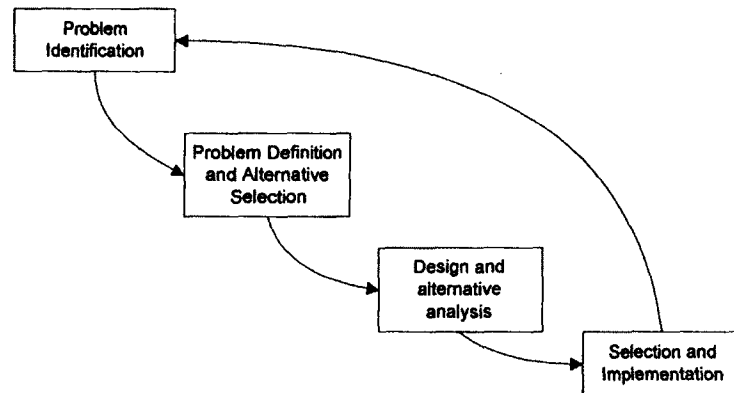


Figure 1. Traditional Decision System development cycle

Note: Adapted from J.P. Shim et al. (2002). Past, present and future of decision support technology. *Decision Support Systems*, 33.

Knowledge Management (KM) presents a comprehensive approach to handle this new challenge and captured the attention of several researchers for a significant period of time as suggested by Raisanen and Oinas-Kukkonen (2008). However, this approach is mostly based on capturing and transferring knowledge from human experts or group of experts, basically organizational learning as noted by Rech, Ras and Decker (2007). Nonetheless, this knowledge usually is captured in the form of explicit or tacit knowledge, rather than integrating implicit knowledge that could be captured from existing automated sources inside a company.

A new paradigm based on knowledge in decision making systems is needed, as noted by Shim et al. (2002), and the incorporation of OLAP, Data Warehousing and Data Mining in this development cycle simplifies the introduction of new decision systems by enriching the view of information on the user side as noted by the authors, and also improves decision making products and processes as noted by Gorla (2003). This new

development paradigm (see Figure 2) is crucial in order to obtain a sustainable advantage, and should provide a clear understanding of how data is used and how business intelligence is captured and integrated inside a company as indicated by Ma et al. (2000), and Moeller (2001).

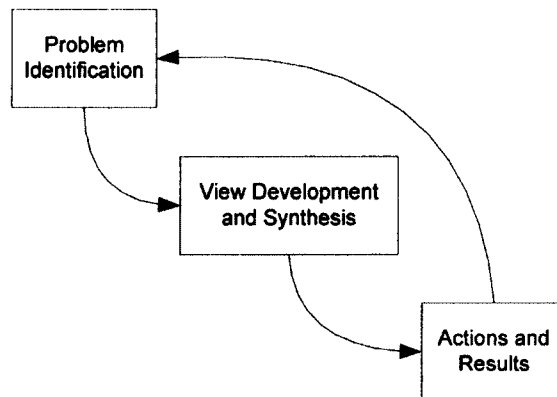


Figure 2. New decision paradigm in Decision making system development.
 Note: Adapted from Shim et al (2002). Past, present and future of decision support technology. *Decision Support Systems*, 33.

Nonetheless, limitations and issues in Data Warehousing, OLAP, and Data Mining still persist as illustrated by Pedersen et al. (2001), Espil and Viesman (2003), and Jensen, Moller, and Pedersen (2001) and represent an interesting field for development of techniques that combine other disciplines such as KBS for a complete solution in this field and to extend the functionality of these tools and integrate the necessary business intelligence in decision making processes.

On the other hand, KBS are not a silver bullet for all decision making problems and also present limitations and problems that demand innovation from the AI

practitioner. As an example of these limitations, the transition from real world characteristics into abstract structures that encapsulate expertise, a process with origins in early Expert System developments, remains far from being a smooth or direct process. Multiple alternatives for developing conceptual or specification models in KBS have been developed, such as KARL or ML as representation languages for the KADS paradigm (Valente, Breuker, & Van de Velde, 1998), SACHEM (Le Goc, Frydman, & Torres, 2002), Modal Change Logic (Fensel, Groenboom, & Renardel de Lavente, 1998), KAOS and Protégé (Gennari et al., 2003; Dieste, Juristo, Moreno, Pazos, & Sierra, 2002) or Constraint Logic Programming (Abbass, Towsey, Finn, & Kozan, 2001). Each of these alternatives indicates the importance of the initial stages of Knowledge Based (KB) development and their significance in the development process. However, little attention has been focused on related data obtained during the knowledge elicitation or knowledge acquisition stages, or in knowledge structures that generate new structures or instances. These latter structures or meta-model of knowledge (knowledge about the knowledge) have a significant impact in the development of new systems, assisting in the design and validation processes (Plant & Gamble, 1997), and may provide an alternative for reinforcing other areas and disciplines for commercial applications, such as OLAP, Data Warehouse, and Data Mining applications.

This study introduces a technique that uses meta-structures of knowledge in advanced decision making systems that solve manufacturing problems and incorporates new knowledge obtained from OLAP, Data Warehousing, and Data Mining. It comprises

technologies such as knowledge engineering, machine learning, and multidimensional logic to provide a comprehensive solution for manufacturing related decision problems.

A practical application of the proposed technique in remanufacturing operations is presented, verifying and validating its results through a comparison of its responses with those provided by a non altered KBS, and responses obtained from other methods in remanufacturing processes in the woodworking industry.

Statement of the Problem

The problem addressed in the current research is to explore the use of meta-structures and higher order knowledge representations to enhance the decision making process in advanced Decision Support Systems (DSS) and integrate knowledge obtained from Online Analytical Processing, Data Warehousing, and Data Mining. Specifically, its goal is explore how new knowledge obtained through Data Mining automatic learning techniques can be integrate into an existing knowledge based solution.

In order to achieve this objective, multiple sub-problems will be addressed as follows:

1. Explore the use of meta-structures and meta-models of knowledge representation as an alternative to traditional methods in AI, such as the Knowledge Acquisition and Documentation Structuring (KADS) and derivatives, incorporating business intelligence on decision making systems and measure the effects of this proposed technique in terms of formalism, validity and verification.
2. Using the proposed technique, incorporate Online Analytical Processing, Data Warehouse, and Data Mining into a decision making system for remanufacturing

problems in the woodworking industry, through the use of meta-structures that capture business intelligence from stored data.

3. The development of this technique that incorporates business logic and data usage into a decision making system will comprise additional questions related to validity and reusability of knowledge in a restricted domain, and automated knowledge acquisition. Thus, the following aspects will be included in the study:

- Techniques for verification and validation of new knowledge obtained through this technique,
- Identify key factors using meta-structures of knowledge that allow the integration of new knowledge from automatic learning techniques.
- Evaluate the obtained system with an equivalent system obtained using existing methods used in KB development, and that do not incorporate new knowledge from OLAP, DW and DM. A particular technique such as KADS and derivatives (commonKADS) for knowledge integration in decision making will be used in the analysis and degree of formalism of the obtained solution.

Statement of the Purpose

The current study explores the use of meta-knowledge representations in the integration of new rules obtained through Data Mining and Online Analytical Processing techniques in an existing knowledge based system. The research focuses on remanufacturing operations in the wood working industry as a case study. To attain this objective, a knowledge based system is modified to incorporate new rules and validate their applicability using meta-rules. The validation process for new rules are compared with other formal methods of knowledge validation, specifically commonKADS, and the modified system results are compared with a non changed knowledge based system.

Remanufacturing operations are selected since they typically address complex processes with high volume of data, multiple views, granularity, integration of data from multiple sources, and dynamic requirements from multiple users to obtain a quantitative objective such as presenting data on multidimensional views (Thalhammer, Schrefl, & Mohania, 2001; Pendse, 2001).

Statement of Need

Current businesses can learn and obtain competitive advantage from collected and stored data usually organized on a database system. Information is presented and flows through transactions and processes that support them. These transactional processing systems (TPS) are usually based on a relational model, and present limited capabilities for large aggregations of data from multiple sources, limited processing capabilities for multidimensional views and demand significant effort to create custom made reports. Furthermore, most reports have limited customization characteristics and present the user with restricted views and data queries, and lack the necessary processing power for multidimensional view and queries of data (Ma et al., 2000; Pedersen et al., 2001). Therefore, the authors suggest the use of advanced DSS tools such as OLAP, Data Warehouse and Data Mining as a way to achieve competitive advantage. Although data precision and presentation are important to achieve this lead, new methods to capture, store and integrate knowledge, intelligent data and business queries are needed as suggested by Nemati, Steiger, Iyer and Herschel (2002). Furthermore, Back, von Krogh, Seufert and Enkel (2005) indicate that an emphasis on administration of existing

knowledge and the ability to constantly create new knowledge moved into products and services as the ultimate key for this competitive advantage.

The process of integrating knowledge and data is not a simple series of steps, and entails a sort of projection over knowledge based on facts as noted by Hoffmann (1998). The author suggests knowledge that describes what should be done with knowledge about facts as a solution for this projection to improve current KBS. However, existing KBS methods present several restrictions with increasing complexity problems in knowledge domains as indicated by several authors (Fensel et al., 1998; Hamfelt, & Hansson, 1991). These restrictions are based on structural problems derived from the specific technique used, such as complications in capturing deep knowledge, inability to provide expanded explanations, or fragmentary nature of knowledge in realistic domains according to the authors. Therefore, a new technique is required to incorporate new knowledge obtained from business processes to improve current solutions. Plant and Gamble (1997) suggest any possible solution will require being an abstract construct, which copes with dynamic functionality of the system.

Knowledge Management (KM) techniques partially address knowledge and business processes integration, focusing on capturing and diffusing tacit and implicit knowledge inside organizations as noted by Spiegler (2000), Alavis (1999), King, Marks and McCoy (2002). However, significant emphasis in research is focused on Newell proposition of knowledge incompleteness characteristics and in the need of transforming knowledge into formal structures as a limitation of knowledge completeness, a scope that may reach beyond KM (Le Goc et al., 2002). Nettleton and Muñiz (2001) noted that KBS

development process requires successfully transforming a conceptual model from a knowledge level into an implementation specification on a symbol level. KADS and Software Engineering do not address this transformation issue directly, and additional information is needed to improve the implementation process and validate knowledge structures, an opportunity where Meta knowledge can significantly contribute to this process and results according to the authors. Nonetheless, even considering knowledge transformation techniques, Wiig (1999) indicates that reconstructing knowledge for a KBS still implies expressing this knowledge in a representation that supports reasoning methods employed by the KBS, and when the source is a knowledge discovery in database technique such as Data Mining, then significant manual interaction is still required. Therefore, new techniques need to be explored to improve the process of transforming and diffusion of knowledge inside organizations, specifically when incorporating Data Mining, OLAP and data warehousing results in existing decision making systems.

This integration between two apparently disconnected technologies as knowledge based systems and data warehousing represents a valid solution for diagnostic and control of production systems in general, and dissemination of knowledge in remanufacturing operations in particular as suggested by Walter (2003) and Ma et al. (2000). In effect, KB can significantly contribute to solve some of the restriction of current DSS technologies, extending its support and enhancing its managerial impact by integrating explicit and tacit knowledge captured by KM technique, and enhancing the decision making process using multidimensional analysis tools; incorporating new patterns, trends and

relationships among existent variables obtained through Data Mining as suggested by Shim et al. (2002), Walter (2003), and Thalhammer et al. (2001).

Characteristics of the problem domain for diagnostic and control of remanufacturing operations in a wood working company as a sample of the proposed technique require the use of advanced decision system tools, such as OLAP, data warehousing and Data Mining. These tools together with web technology are components of four key technologies in Decision Support Systems that influence current and future developments, as noted by Shim et al. (2002) and have been successfully integrated in a data webhouse as presented by Moeller (2001), and Kimball and Merz (2002). The authors suggest the use of a collection of databases recording user activity on distributed servers allow gathering information about user behavior on distributed environments with concurrency being tracked on multiple servers that process user related information, as indicated by Moeller (2001). This integration of multiple technologies together with distributed multi-databases converge on developing intelligent distributed systems and also contribute to enhance creation, transformation and diffusion of business intelligence on everyday processes as suggested by the authors. Thus, this integration capability is a necessary characteristic of the application of the technique in a restricted domain.

Initial efforts on integrating knowledge systems and Data Warehouse through a comprehensive class designed to provide users with an environment for decision making are reported by Nemati et al. (2002). Nonetheless, Thalhammer et al. (2001) and Shim et al. (2002) noted the use of active Data Warehouse principles to achieve this objective as well. By extending the concept of Decision Support Systems (DSS), the authors suggest

multiple forms of integrating business knowledge and data processing revealing a diversity of techniques on this field. On one hand, knowledge based principles are applied to enhance results obtained using pure DSS techniques, such as software agents and rules that capture the necessary logic of the decision making process, handle the complexity of information overflow, revisiting hierarchy aggregation on an OLAP environment, and allow the creation of intelligent Data Mining agents. On the other hand, a complete integration of these technologies is proposed to create an extension of Data Warehouse into a conceptual domain, through the development of knowledge warehouses. This new category of intelligent DSS allows current knowledge management paradigm to evolve over time and enhance knowledge creation, transformation and diffusion on a dynamic way.

In conclusion, the proposed technique represents an innovation in this field, since it is oriented to integrate of KB solutions with other DSS technologies, and verify and validate the technique results at a conceptual level. The development of solutions obtained with this technique throughout knowledge based systems principles that incorporate knowledge and non-quantitative reasoning appears as a logical step toward a more comprehensive solution for Data Warehouse, Data Mining and OLAP decision problems, specifically in remanufacturing operations of a woodworking company and also represents an innovative case for the applicability of the technique.

Statement of Research Questions

The current study addresses the following research questions:

1. Can a technique that integrates knowledge meta rules assist in automating knowledge transformation and contribute to reduce the radical incompleteness at knowledge level assumption, in terms of Newell's proposition?
2. Does the use of meta-knowledge representation which incorporates Data Warehouse, OLAP and machine learning tools provide a feasible solution as an AI technique in enhancing DSS functionality on a real manufacturing problem?
3. Do solutions obtained using the proposed technique, encapsulate verified and valid knowledge?
4. Can meta-knowledge, Data Warehouse, OLAP, and Data Mining integrate into an active Knowledge Management System and effectively assist remanufacturing related decisions as in the case of a woodworking problem?

A solution for remanufacturing operations for a woodworking company will be used as a case sample.

In terms of research hypotheses, these research problems can be formulated into two different areas, as follows:

For the proposed technique, the following hypotheses are generated:

1. By incorporating meta-knowledge representations, a new technique that extends OLAP, Data Warehousing and Data Mining can be established by meeting goals and requirements of knowledge warehousing as noted by Nemati et al. (2002).
2. Metarules can enhance the responses of a DSS if compared with a non modified DSS system, and present a similar formalism in terms of knowledge incorporated on the system if compared with an equivalent solution obtained using the KADS model for KBS development.

For the application of the paradigm on a restricted problem domain, the following hypothesis is formulated:

3. Integration of meta knowledge representations with DSS tools improve solutions provided by DSS alone, addressing eleven requirements that multidimensional data must meet in new application areas, as suggested by Pedersen et al. (2001).

Assumptions

The study assumes the following aspects:

1. Interaction between problem solving methods and domain knowledge is required, as noted by Valente et al. (1998). Therefore, the structure of domain knowledge is constrained by inherent characteristics which need to be represented in the symbol level.
2. In the development of an application obtained using the proposed technique, knowledge can be established in terms of rules that denote an absolute inference rule, in terms of $P \rightarrow Q$. No gradual knowledge will be addressed, and proposed solutions obtained through this technique will not contain topos (Dieng, Corby, & Lapalut, 1995) as explanatory instruments or during the development process.
3. The case limitation into a particular problem domain such as remanufacturing on a woodworking company, in the application problem domain of the technique does not restrict the results in terms of correctness of the Knowledge Based System that integrates with the Decision Support System.

Delimitations

The study presents the following limitations:

1. In the formulation of the technique, rules that capture meta-knowledge representations will be preferred rather than other formal methods for representing

reasoning modeling, such as frames, formal specification languages (Craig, 1991), or modal change logic (Fensel et al., 1998).

2. Results provided by the KBS obtained using the technique do not implement optimization based solution but focus on providing additional information related to knowledge incorporated on the solution.
3. Closed world limitation. No further data analysis or Data Mining is performed by the system beyond the scope of the problem domain.
4. The system is implemented using meta-knowledge logic principles. Higher level representations are used at conceptual and design level, for implementation purposes knowledge will be encapsulated using production rules on a recursive basis.

Definition of Terms

Artificial Intelligence (AI): a discipline inside computer sciences that allow machines to do things that people think require intelligence or that would require intelligence if performed by a human (Jackson, 1985). The term is also referred as the science of mimicking human mental tasks on computers (Hopgood, 2003).

Architecture: In Knowledge Based Systems, corresponds to components and their interaction that comprise the final system. (Oussalah, 2003)

Components: Computer related elements and their storage with a single or multiple interfaces that represent a system. (Oussalah, 2003)

Data Warehouse: Large database that summarizes information from multiple sources. It comprises pre processed information stored in current detailed, older detailed, lightly summarized, highly summarized data, and metadata. It can be used by businesses to obtain a competitive advantage based on stored information, allowing decision-makers with faster access to widespread data from a unique, usually centralized repository

(Moeller, 2001). The subcomponents of a Data Warehouse, as suggested by the authors include: (a) A source or multiple sources from which data is extracted, (b) a processing software, called middleware, that process and translate data from diverse inputs and populate the Data Warehouse, (c) a storage structure, (d) and desktop tools for querying and reporting as noted in Figure 3. A key component of Data Warehouses is the transformation that takes place in the middleware component, which can be field translation or reformatting, data alteration, aggregation or summarizing, and reformatting of data structures as noted by the author.

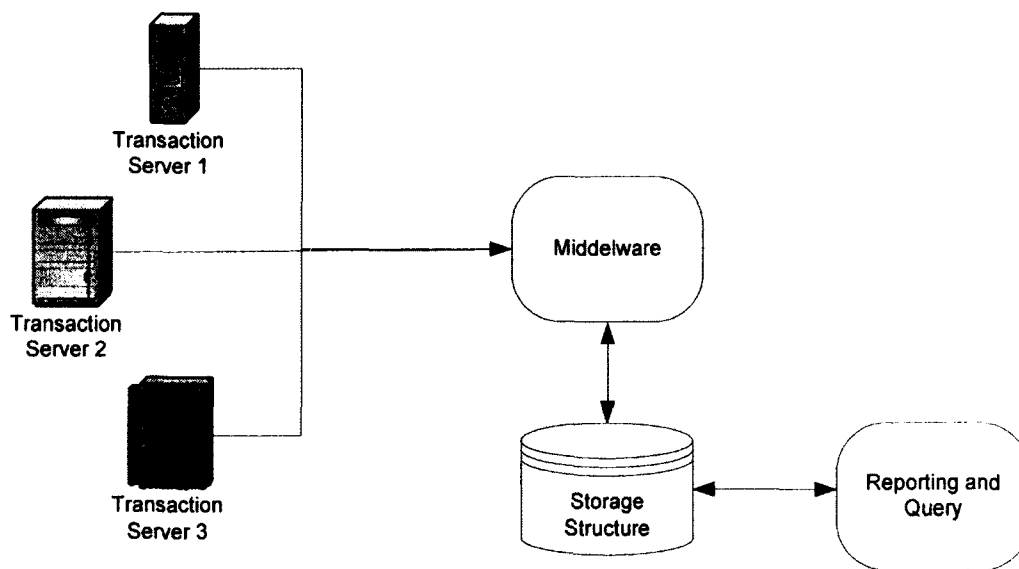


Figure 3. Data Warehouse components.

Note: Adapted from Moeller (2001). *Distributed data warehousing using web technology*. NY: Amacom.

In terms of overall functionality, Thalhammer et al. (2001) indicate that Data Warehouses extract, transform and load (ETL) data from transactional sources without user intervention.

Data Warehouse bears a close relationship with OLAP and Data Mining being referred as a significant component of contemporary decision support systems paradigm as presented in figure 4 (Shim et al., 2002). Obtaining consolidated and pre processed information, OLAP enhanced reporting capabilities represent a natural component and Data Mining provides new relationships among existing data. Their integration on multiple applications includes resource planning, sales analysis, budgeting, and production forecast (Gorla, 2003).

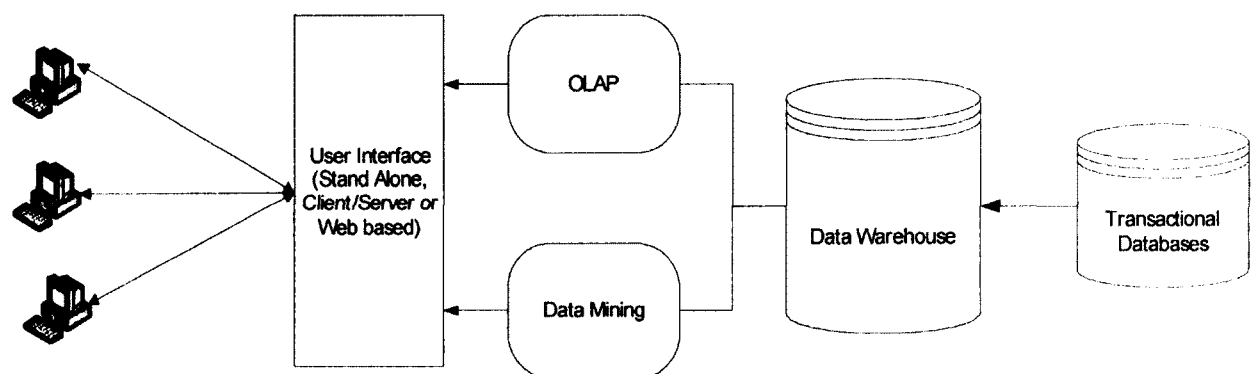


Figure 4. Relationship between Data Warehouse, OLAP and Data Mining on current DSS paradigm.

Data Mining: Process of applying artificial intelligence techniques to data sets for the purpose of determining patterns and allow predictions of the system (Ma et al., 2000). Among the multiple types of Data Mining tools available, the authors suggest a taxonomy which includes the following types: (a) Multidimensional databases with learning capabilities, (b) Statistically based applications, which comprise advanced inference techniques such as cluster analysis and regressions, and (c) Artificial Intelligence based solutions that employ heuristics and advance approaches to determine new relationships, patterns, and trends of collected data. Its functionality includes data pattern determination, format capabilities, analysis of data content, and data synthesis. Data Mining is vastly used in resource planning, customer relations, and supply chain related applications.

Decision Support Systems (DSS): Computer programs used to support complex decision making processes. It incorporates powerful database capabilities, a management modeling functionality, and reporting components for interactive queries. DSS found its origins in efficiency principles of early database designs during the 1970s. Its latest developments integrate this technology with web functionality, allowing managers to capture and model their decisions beyond the borders of their physical location (Shim et al., 2002). The concept derives from decision theory principles and database foundations.

Expert System (ES): Computer program that can solve problems the way human experts will do. The ES domain of competence is narrow, specific, and includes areas where tasks are not structured, deals with incomplete data, or do not exist an algorithm to provide an acceptable solution (Meseguer & Plaza, 1992). An ES manipulates symbols,

can explain a conclusion, and advise on planned situations. It may also be referred to as a subset of Knowledge Based Systems or Intelligent Systems, although Benfer, Furbee and Brent (1996) noted that these terms can be used indistinguishably. Most ES implement first order logic representations, although restrictions in using FOL are founded in the fact that almost all generalizations have exceptions, or can be held true to some degree (Russel & Norvig, 2003). Furthermore, Chandrasekaran, Johnson and Smith (1992) suggest a definition for Knowledge Based Systems as an explicit representation of knowledge with an inference process that operates towards a defined goal. The inference process consist of a series of steps, each of them generating knowledge that repeats until it fulfils requirements of the problem solving goal, a definition that resembles one for ES.

ES development cycle: Methodology for the transfer of knowledge from a human expert into a computer program (Nabil, 1999; Jackson, 1999). Critical steps in development time are Conceptualization and Formalization, as noted in Figure 5. For most other stages, techniques have been developed to improve their development time and its throughput.

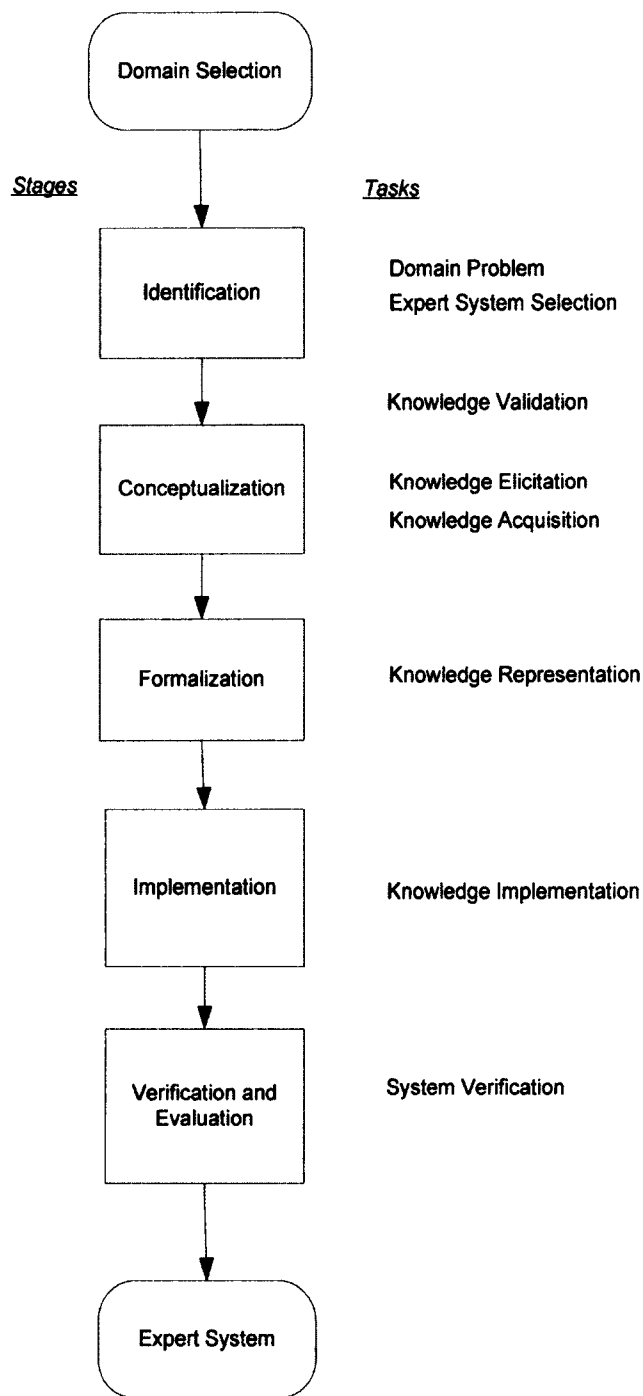


Figure 5. Expert System Development Cycle

Note: Adapted from Nabil (1999). Development of an Expert System. *The State University of New Jersey Rutgers Website*. Retrieved February, 1999: <http://cimic3.rutgers.edu/~adam/courses/es/node8.html>.

KADS methodology: Mostly used for system specification, it corresponds to a framework for development of KB applications, which suggest general guidelines to build several models that reduce differences in required and final system behavior. It evolved into commonKADS, a comprehensive and widely used methodology for KBS (Akkermans, Van de Velde, Wielinga, & Schreiber, 1993). The process starts with an organizational model, which indicates the category of problem related to the complete organization that the system will solve or assist, an application model which specifies the required behavior in terms of behavior with other systems and restrictions, a task model that indicates the required tasks that the system will perform, a conceptual model that is translated into a specification model, and finally a design model before its implementation and coding stage, as noted in Figure 6. (Le Goc et al., 2002) The conceptual model is comprised of three layers: The domain layer, the inference layer, and the task layer. The main objective of KADS is to cope with knowledge complexity by dividing the general model to be implemented in 6 sub models: Organizational, Task, Agent, Communication, Design, and Expertise models. The development methodology allows the description of problem solving behavior at an abstract level, without representation or implementation details.

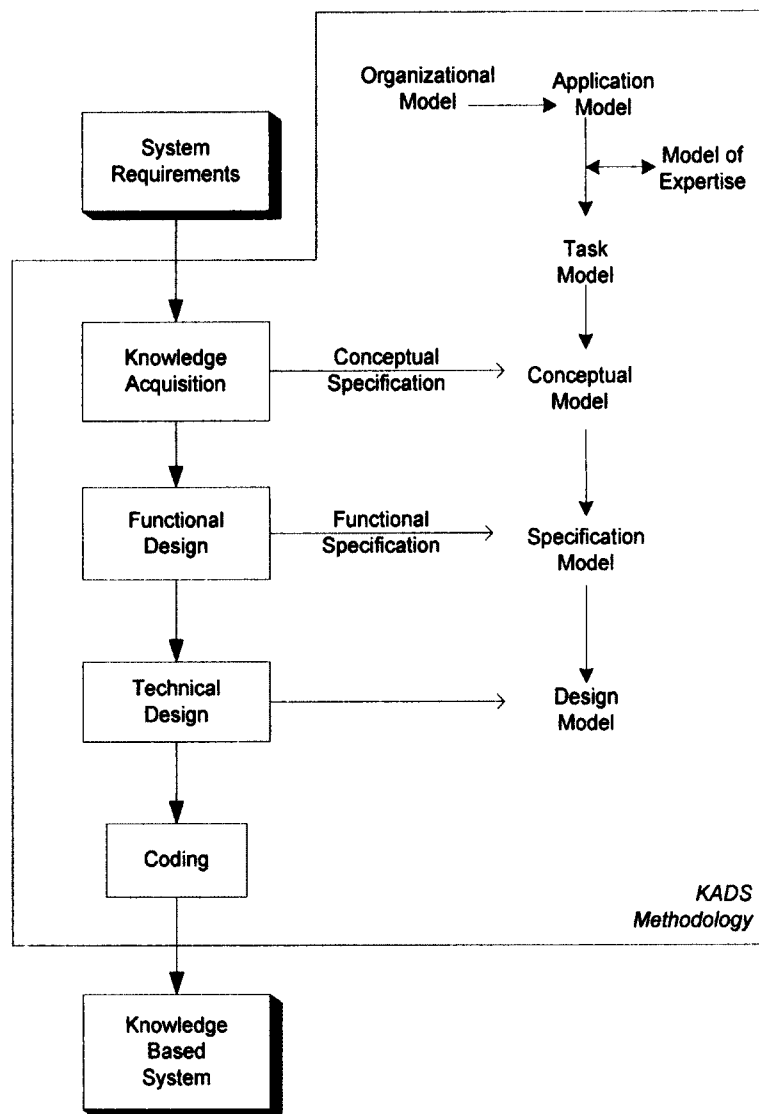


Figure 6. KADS methodology stages

Inference engine: Main component of an ES that provides the mechanism to achieve a predetermined conclusion and strategies to represent heuristics, as presented in Figure 7. An inference engine represents a repository for problem solving strategies, a required component of ES as noted by Hoffmann (1998).

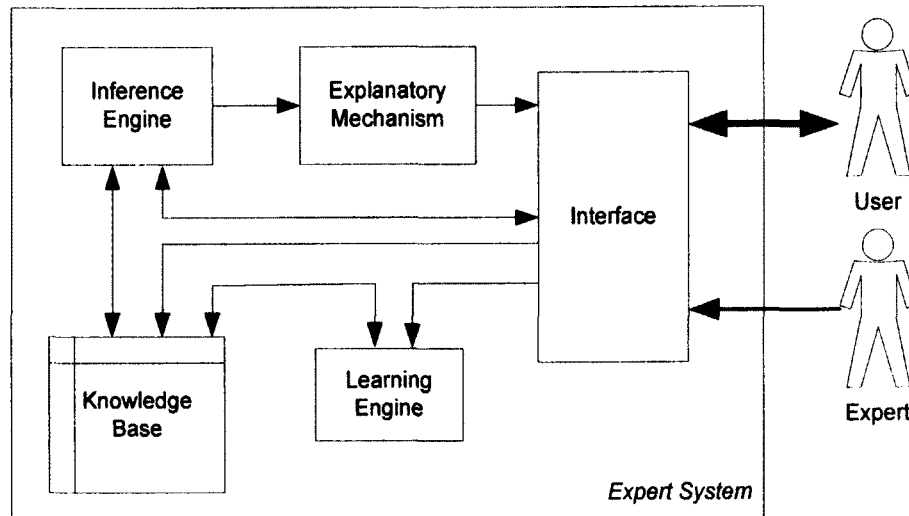


Figure 7. Simple model of Expert System Components.

Heuristic: Map of strategies that the expert has to perform a non-structured task, which may not be necessarily proved correct but simplifies the process of solving a problem. They are usually expressed in terms of a rule of thumb. (Jackson, 1985)

Knowledge: Justified true belief. Believing in something for a justifiable reason and if it also becomes true, then you know it (Russel & Novig, 2003). In small domains, representation of knowledge is not an issue. However in more complicated problems, a more flexible and generic representation is required. These knowledge representations are comprised of actions, time, physical objects and beliefs. Several definitions of the term range from conceptual to philosophical, and small to large scope as noted by Beckman (1999), from information used in problem solving to reasoning about information and data to allow problem solving or decision making.

Knowledge management: Discipline that studies how businesses incorporate knowledge in their creation of goods and services (Matison, 1999). It comprises an understanding of the value chain, or how the business is organized in terms of its functional components and how it delivers value to their customers; a value proposition in terms of the product or service and standards under which the business commits with the customer in exchange of money or compensation; modeling and visioning, which relates to the what the business will do; knowledge networks, basically individuals in an organization that share the same interest in some set of knowledge; and knowledge economics, or how the organization value and trade knowledge.

Business knowledge is defined in terms of data, context and their application to specific business objectives.

In terms of formal criteria, KMS can be classified in one of the following options:

- Decision support systems.
- Computer system developed in support of value chain, value proposition, modeling and visioning, knowledge networks and economics.
- Any computer system that integrates data + context + application to the solution of business problems.
- System identified by a common objective, such as a collection of unrelated software applications that define a group of knowledge body inside an organization.

It also comprises multiple definitions with variable scope, from capturing collective expertise on an organization to a systematic approach to capture, transform and use knowledge to create value as suggested by Beckman (1999).

Online Analytical Processing (OLAP): A software class that allows fast, consistent, consolidated, and interactive access to multidimensional views of information, and allow customization for analysts, managers, and executives inside an organization. Data is represented through facts, which are perceived as points on a multidimensional space. It is referred as an evolved version of the relational model for data modeling (Thalhammer et al., 2001; Espil & Vaisman, 2003).

OLAP technologies are closely related to cube concepts, a representation of multidimensional analysis. The data structure that groups measures and dimensions in OLAP is the cube, as illustrated in Figure 8.

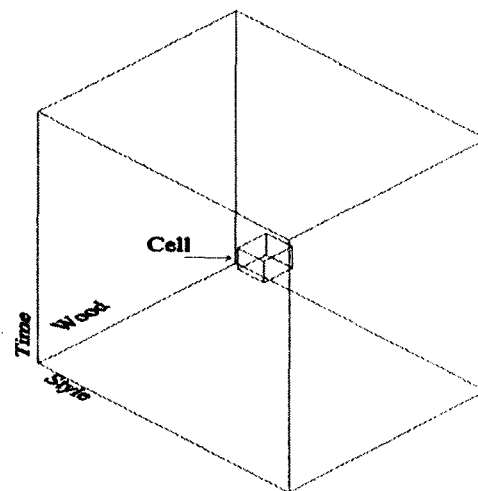


Figure 8. OLAP cube representation

Each cube plays a similar role as databases in transactional processing. Two basic operations for data aggregation are provided: Drill down and slice and dice. Drill down allows detail information and consolidation of data inside a dimension, while slice and dice allows regroup of dimensions.

Nonetheless, two different approaches for OLAP implementation are commonly adopted: MOLAP and ROLAP. MOLAP aggregate and clears data in multiple dimensions, but requires that the cube must be updated periodically. ROLAP, on the other hand does not requires data refreshing since data comes directly from original sources but relies on indices built at tables for data access. Nonetheless, a market preference toward MOLAP indicates that ROLAP has not achieved a necessary level of maturity for commercial applications (Thomsen, 2002; Jensen et al., 2001)

Paradigm: Conceptual framework of ideas which explains how they relate each other. It also may be used to refer to a model which represents something that is considered the basis of a methodology or theory (Encarta encyclopedia, 2003)

Validation: Phase of the ES development cycle, which ensures that the right ES has been built, by analyzing results and functional aspects of the knowledge and expertise that are implemented in the system. It corresponds to build the right system, as suggested by Mosqueira-Rey and Moret-Bonillo (2000). As a difference with conventional software, validation starts during the development of the ES instead of the end, when testing occurs in conventional programming methods as noted by Juristo and Morant (1998).

Verification: In conventional software, it corresponds to the process of testing on a specific stage that a system fulfils the requirements established during the previous phase (Juristo & Morant, 1998). In ES it corresponds to ensure that the solution contains no errors and includes most of the detected specifications. ES cannot establish initial requirements due to the nature of knowledge elicitation. In other terms, verification is the process of certifying that the right system was built according to specifications. (Mosqueira-Rey & Moret-Bonillo, 2000)

CHAPTER II

REVIEW OF RELATED LITERATURE

Knowledge Based System Development

The development processes of KBS and conventional software have several concepts in common, as noted by Juristo and Morant (1998). Both categories of software generated multiple models based on requirements established by external entities to the system, and both are required to formally verify and validate their results. However, significant differences among the development process entitle each of these categories to specialize their models according to the emphasis that should be established in their specific domain. The generation of separate conceptions in software have been justified through identification of differences among categories: Intrinsic characteristics, such as the specialized knowledge that ES addresses or explicit representation of domain knowledge in KBS; and operational reasons, such as the impossibility to establish a criteria to determine correctness of a system in the beginning of KBS and its changing nature during the development process, or differences in validation of conventional systems which is performed when its implementation is completed in opposition to the development of KBS, where validation takes place during the development of the system, as noted by the authors. Therefore, a specialized model is required to guide the development of a KBS different than those available in conventional software.

Early Knowledge Systems development methods focused on selecting and extracting expertise from human experts, and its completeness in terms of structured knowledge remained under permanent discussion as illustrated by Hoffmann (1998) and

Wielinka, Akkermans, and Schreiber (1993). Implementation emphasis is a characteristic of early methods, according to the author. Second generation ES focused on knowledge and the introduction of the KADS methodology represented a significant contribution in this aspect, as suggested by Fensel et al. (1998). The authors noted that benefits from this methodology are the distinction of different levels of abstraction and identification of several models that cover aspects of the development process that were not addressed by its predecessors. It also provides guidelines for establishing a conceptual model, which represent an informal representation described in an implementation independent form, a desirable feature on early stages of development (Le Goc et al., 2002). It also differentiates a conceptual model from specification aspects, which are appraised through the identification of three different levels of knowledge: Task, Problem Solving Methods (PSM), and Domain knowledge. However, the authors suggest significant deficiencies in KADS, in part due to the elusive nature of a definition of the concept of knowledge. This restriction in modeling suggests that a division of a general model into different derived models that interact is required; on a divide for conquer approach. Le Goc et al. (2002) also indicate that conceptualization, specification and modeling should be explicitly related in KADS in order to facilitate its computational implementation, and knowledge required for the construction of the system should be modeled and included. This latter diagnostic and suggestion induce generating a metadata based model that incorporates these meta-knowledge principles.

A derived methodology, commonKADS was proposed to allow reusability of knowledge components among similar problem classes. This methodology suggests the

specification of a library organized on three basic structures: Task, Problem-Solving Methods (PSM), and Domain knowledge (ontologies). Only five models are suggested on commonKADS, which addresses part of the problems to implement the general KADS model. Its model set includes organizational, task, agent, expertise, communication and design models as noted by Speel and Aben (1998). Models in commonKADS can be grouped in feasibility, knowledge modeling and design and implementation stages, as noted in Figure 9.

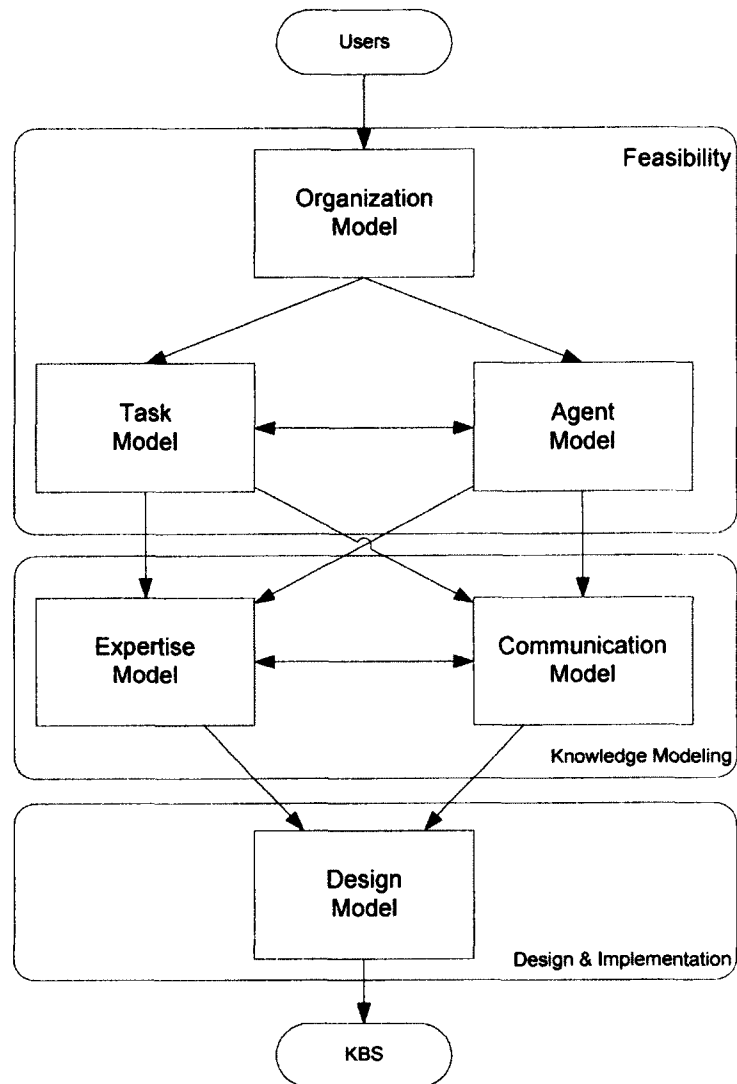


Figure 9. CommonKADS model.

Note: Adapted from Speel & Aben (1998). Preserving conceptual structures in design and implementation of industrial KBSs. *International Journal of Human-Computer Studies*, 49, 547-575.

Although this model set has received better acceptance, being adopted as the basis for many derived models (Hicks, 2003; Oussalah, 2003; Le Goc et al., 2002, Fensel et al., 1998; Haouche-Gingins & Charlet, 1998), some additional aspects need assistance in order to make the model operative, as described by Valente et al. (1998). The authors

established a list of areas for improvement in KADS and commonKADS, being the most relevant aspects the intractability of the library in the case of a large number of PSMs, requiring additional information for selecting the appropriate option to be added into the library and common characteristics were identified which were not reflected on the library structure; Reusability of the components, with multiple modifications such as Protégé (Gennari et al., 2003), component based approach (Oussalah, 2003), or SACHEM (Fensel et al., 1998).

Protégé is a tool that initially was oriented to help users to build a customized tool that assist in knowledge acquisition for specific application areas (Gennari et al., 2003), and evolved into a general environment for knowledge modeling as suggested by the authors. Its early releases were oriented towards allowing knowledge engineers build Opal-like tools for every domain. A meta-tool generated knowledge acquisition tools from structured concepts. Its main concept is that every stage in development generates meta-knowledge for next stage that helps reduce barriers in knowledge acquisition. Although other methods were concurrently developed as noted by the authors, such as Expect and Mole, they were mostly oriented to complete and correct the user knowledge base instead of assisting in the development process itself. The addition of problem solving methods (PSM), general algorithms used in the solution of different tasks, allowed Protégé to evolve and include reusable methods extending the initial unique ESPR method available in its first versions. Figure 10 illustrated Protégé general usage and its interaction with knowledge engineers and domain experts through specific knowledge acquisition tools, and end users with the resulting Expert System.

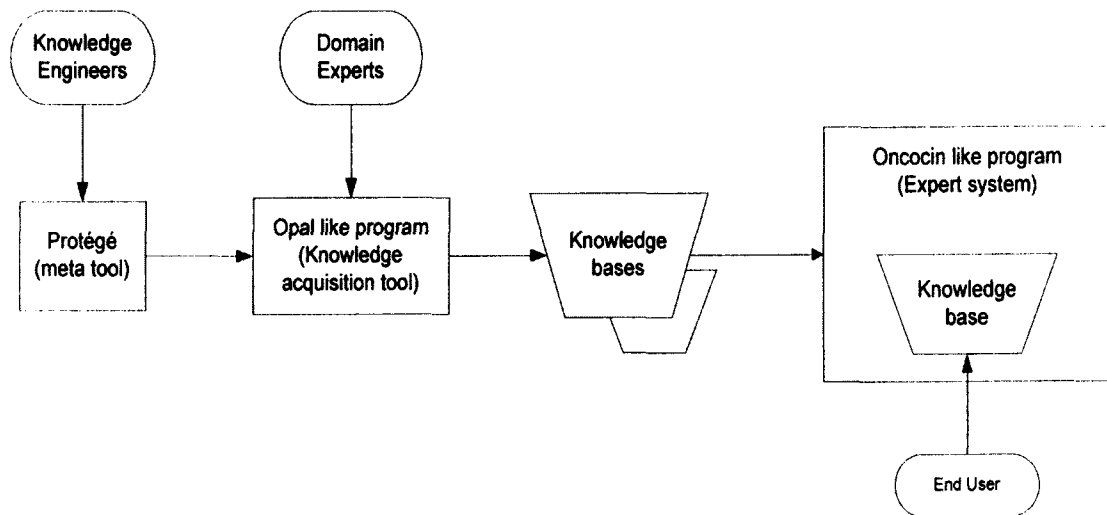


Figure 10. Protégé general usage.

Note: Adapted from Gennari et al. (2003). The evolution of Protégé: An environment for knowledge-based system development. *International Journal of Human-Computer Studies*, 58, 89-123.

Finally, Valente, Brueker, and Van de Velde (1998) suggested that PSM cannot be completely described without referring to knowledge domain characteristics. If this observation is accepted, metadata and meta-knowledge techniques can contribute to specify these missing characteristics on the commonKADS model.

Ontologies emerged as a by-product of KADS methodologies that acquired its own dynamics, being used in similar terms as objects are used in traditional software. These models of bodies of knowledge that describe concepts, domain, and their relationships transcended into other areas of Artificial Intelligence, such as agent-based software, knowledge acquisition, and natural language processing as suggested by Cranefield and Purvis (1999). It helps researchers defining a common set of terms that

they can share; allow reuse of domain knowledge, and separate operational from domain knowledge as suggested by Noy and McGuiness (2001). It also can be described as a set of concepts in a domain, also called classes or concepts, which contain roles or properties that describe characteristics of these concepts, and their restrictions. In ontology terms, a collection of ontologies and their instances constitute a knowledge base, as noted by the authors. They are structures on a hierarchy of ontologies, and their development corresponds of an iterative process of identification of these concepts and their characteristics. The representation of ontologies will depend on the knowledge representation selected, which can be rules, frames or semantic networks as suggested by Jackson (1999). The standardization of their use through the adoption of standards such as UML notation (Cranefield & Purvis, 1999) suggests that their influence into DSS and Data Warehouse technology can be significant.

Other methodologies for KBS development are based on mathematical principles, and may be used in conjunction with more comprehensive models to supplement their deficiencies, such as meta-representation model (Abbass et al., 2001) for specification modeling, LOUIS system for expertise transfer (Heliades & Edmonds, 1999), Metalogic principles for stratified knowledge domains (Hamfelt & Hansson, 1991), COMPLAN, a model based on hierarchical planning methodology and Generic Task Toolset (Martinez-Bermudez, 2001) for manufacturing process planning.

Abbass et al. (2001) illustrated a method to integrate operational research solutions with artificial intelligence principles into a decision support system using constraint logic programming as the nucleus or kernel of an integrationist model.

Although the method is explained in terms of a formal meta-language definition, the authors emphasized optimization results and lack implementation details in the description of the interface of each component. In a similar study, Hicks (2003) successfully integrated business Knowledge Based Management principles into an extended commonKADS type, three-level model. The author extended his model into a complete development cycle. Valente et al. (1998) proposed the development of a commonKADS library of general problem solving methods, indexed by taxonomy of task types. Each element of this library is stored in a domain division if related to domain knowledge or in the task division if it related to problem solving knowledge. Each domain division element is classified either as domain ontology that symbolize terminology of domains in structured manner, or a domain model, which represent reusable structures of knowledge that can be used during the problem solving process. Elements in the task division can be either a function, basically functional data-flow or function structures, or group of functions. These elements are connected through problem solving methods (expansion or control methods), control structures and features. This classification allows the use of components on a modular basis for different domains. However, the authors warn about its difficulties on access and applicability in a practical problem.

Finally, Lee (2003) introduced the use of graphical tools, such as information structure graphs as a metadata scheme to improve data management. Nevertheless, these structures may be extended into other domain problems, and assist in capturing meta-information in KBS development.

Meta-knowledge Techniques and Knowledge Management in KBS

Meta-knowledge, a term used for structures about structures or knowledge about knowledge as suggested by Plant and Gamble (1997); and Kalfoglou, Menzies, Althoff, and Motta (2000), corresponds to a concept used for system design that improves the development process of knowledge related software. As noted by the authors, two major aspects specify the type of knowledge that is modeled through meta-knowledge techniques during the development process: A static type approach, and also referred as computational type, which focuses on the structure of knowledge and how it is stored and retrieved on the system. This aspect is closely connected to system verification, as suggested by the authors; and a dynamic or non computational type, which deals with the behavior of knowledge during execution and relates to a validation aspect of development. A third category can also be obtained by combining these two original types, as suggested by Kalfoglou et al. (2000).

The influence of KADS methodology on classification of meta-knowledge structures is illustrated by the authors, which elaborate on ontologies and problem solving methods (PSM) as static meta-knowledge types; and case based reasoning (CBR) as an example of the dynamic category. Moreover, Kalfoglou et al. (2000) provide additional insights of the benefits of using Meta-Knowledge on KBS development, with an emphasis on static types, and reinforce the importance of Ontologies as a common lexicon for all agents involved on the development process. Together with PSM, Ontologies allow knowledge engineering to move from an art status activity into a discipline with a formal process. Nonetheless, the authors also indicated that these two

categories tend to disagree in terms of the impact on the development process, and more knowledge-related metrics are required for an accurate quantification of these benefits.

Herrmann, Kienle, and Reiband (2003) expanded these two categories to operationally evaluate meta-knowledge impact on the development process and evaluate their impact as success factors in knowledge management techniques. The authors suggested six new sub categories, which can be rearranged into the original super classes as follows:

1. Static knowledge: Characteristics related to content, basically timeline and quality of the knowledge included; and structure of content related to internal characteristic of the knowledge, and
2. Dynamic knowledge: Characteristics of the participant, particularly on experiences of other users and knowledge exchange; post usage of entered data, where users consider how the data provided relates to uploaded content; cooperation between participants of the development effort, mainly aspects of the knowledge entered and its relevance in the work of other actors; and self efficacy related to expectations and capabilities provided by the system.

Nemati et al. (2002) without explicitly defining the concept, suggest that meta-knowledge combined with codified knowledge can be used to successfully capture, codify and classify new knowledge, and therefore incorporate it actively on a new system. Nonetheless, the authors identify instances of the knowledge classes, such as production rules, text streams, binary large objects, mathematical models or scenarios but did not elaborate on a further hierarchy neither on a process to explicitly incorporate meta-knowledge on the development process. However, a general model of the

knowledge worker is presented by the authors, indicating four stages of knowledge experience inside an organization, and present the foundations for Knowledge Management cycles: (a) socialization, which corresponds to knowledge diffusion in the form of concepts, heuristics, or conceptual maps; (b) articulation of tacit into explicit knowledge through specifying the purpose and parameters related to a decision, and proposed scenarios for its implementation; (c) Integrating different types of explicit knowledge into new relationships; and (d) internalizing knowledge, which relates to abstract understanding processes. The relationship of these states is illustrated in Figure 11.

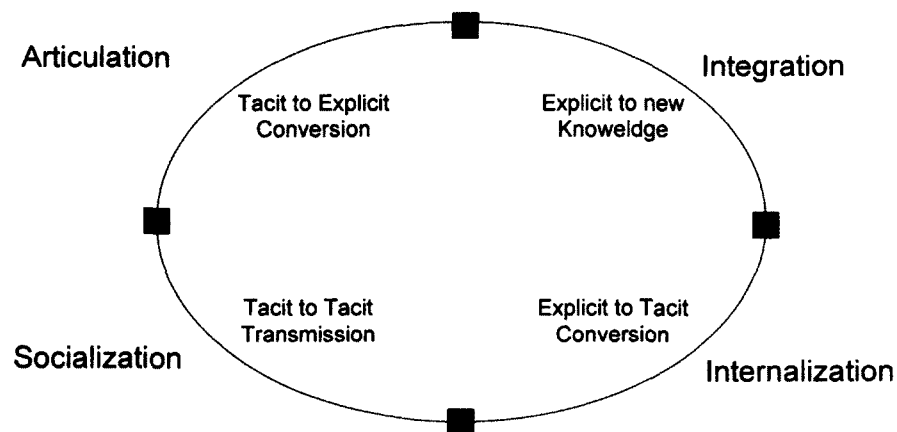


Figure 11. Knowledge management cycle.

Note: Adapted from H. R. Nemati, D. M. Steiger, L. S. Iyer, & R. T. Herschel (2002). Knowledge warehouse: an architectural integration of knowledge management, decision support, artificial intelligence and data warehousing. *Decision Support Systems*, 33. 143-161.

Although a comprehensive model is presented and used as the basis for an extension of the Data Warehouse concept into a knowledge warehouse homology, the authors omitted automated tools for knowledge acquisition, such as Data Mining or statistically related techniques and should be expanded to accommodate these technologies.

The inclusion of meta-knowledge into formal structures is another area where research has developed at a different pace than other aspects of KBS. Dieste et al. (2002) suggested that the selection of a conceptual model (CM), an event with similar magnitude than problem analysis in traditional software development, has a direct impact on how the design (Formalization stage in KBS) will be conducted. An enrichment process of the ontologies detected through the conceptualization is necessary, and the authors suggest two possible mechanisms to achieve this enrichment: Expand the concepts that can be used on the selected CM using special builders or semantic seeds to represent the added concepts; and defining a meta-model, a structure to build individual CMs as sub classes of a super class based model as noted in the knowledge acquisition in automated specification (KAOS) and enterprise modeling (EM) to represent static and dynamic aspects of knowledge. Nonetheless, a procedure to obtain a similar structure for automatically generated knowledge and how to integrate dynamically obtained knowledge representations remains absent in the study.

Fensel et al. (1998) suggested that Modal Change Logic (MCL), a formal specification of dynamic aspects of language related ontologies, provides a mathematical based opportunity for knowledge reasoning modeling. However, the syntax of the

proposed algebra does not supply direct insights about relationships and knowledge structures for the novice knowledge analyst. Therefore, explicit and visual based modeling techniques should be preferred in this field.

Meta-knowledge is assimilated with strategic knowledge by Heliades and Edmonds (1999), suggesting that augmented documentation related to design of a system may improve the development process of traditional software. Although the concept of this documentation structures, called design rationale (DR) may also be extended into KBS, the authors do not elaborate on a conceptual level about these structures but instead describe task level operation of an obtained system that incorporate such structures. Abbass et al. (2001) and Christiansen and Martinenghi (2000) indicate the use of constraint logic (CL) for meta-logic programming should improve the development and integration of other areas of AI or Operational Research solutions by separating objects (Ontologies in KBS) and meta-level structures (meta-knowledge) for further analysis in formalization and conceptualization stages respectively. Nonetheless, its syntax and notation do not provide a direct representation of relationship among ontologies.

Therefore, in order to improve the knowledge included on a KBS application, a step to extend the content of the ontologies modeled on the problem domain is to include a model of meta-knowledge as part of the conceptual model to obtain different views of the knowledge modeled on a similar extent that meta-models enrich ontologies on the development of KBS.

Artificial Intelligence in OLAP, Data Warehouse, and Data Mining

Artificial Intelligence (AI) consists in the development of technologies that can represent knowledge in a similar manner than a human will do, and also are able to translate these knowledge structures into declarative and procedural forms that may be processed by a computer (Jackson, 1999; Openshaw & Openshaw, 1997). AI has been used on several disciplines with relative success, yet remaining research community skeptic about it results. Reich (1996) summarized the expectations that research community made in 1958 about simulating a human brain that will help to understand how complex operations occur. In 1966, other researchers made predictions about the future and how this new technology will modify our life and behavior, although these changes never took place in our society. The author suggests that this phenomenon is partially explained because of (a) lack or partial absence of knowledge in the field, (b) intricate knowledge operations of the task, and (c) continuous development of the discipline that make AI products obsolete or difficult to maintain before the next leap. Tommelein et al. (1992) also suggested that this phenomenon may be explained by understanding how research on AI has been conducted by private corporations in the past, which consider results of this research as proprietary information, and differences on how people address a problem and how models represent it. However, additional considerations about why results in AI have not been as successful as expected might also consider deficiencies in techniques that can be applied to solve a practical problem, because of highly complex methodologies based on theoretical aspects.

Efforts on integrating Artificial Intelligence with Decision related systems

because of the increasing complexity in the type of data analyzed by this type of systems is introduced by Cody, Kreulen, Krishna, and Spangler (2002). The authors indicated the necessity of business executives to integrate knowledge and management, measured through business data, on a comprehensive solution. The authors illustrated on two applications that integrate data and text analysis through OLAP-like reporting capabilities, and suggest the integration of ontologies as a promising area for future development on this field. Although the proposed amalgamation increases the functionality of resulting systems, it does not solve current deficiencies of OLAP or Data Warehousing, and therefore it represents an extension to current functionality but not a complete integration to solve or address current deficiencies and problems.

In the application of AI into OLAP for improvements of current deficiencies, a significant contribution related to automate routinely decisions is introduced by Thalhammer et al. (2001). The authors suggested the usage of rules to improve data quality and automated addition of new knowledge obtained from the Data Warehouse. A novel type of Data Warehouses is defined: Active Data Warehouse. Espil and Vaisman (2003) noted the importance of rules into the definition of new detected hierarchies on OLAP systems, allowing different levels of aggregation and extending hierarchy instances to incorporate exceptions due to uncertainty, unreliable data or specific policies. However, the authors caution about potential performance issues when the revision mechanism is embedded on an OLAP final user tool.

The Active Data Warehouse concept is presented by Thalhammer et al. (2001) and relates to the passive role of conventional Data Warehouses. The authors extended the functionality of Data Warehouses towards incorporating analysis rules for data analysis and decision making functionality. These analysis rules ultimately replicate tasks that an analyst will perform, especially related to routine and non routine tasks in the form of events that occur, with specific conditions that holds and then an action is performed. Each analysis rule is associated with a set of cubes and each cell of these cubes is evaluated against two conditions. A decision consist of executing a transaction on primary dimension level, initial further manual decisions or perform no action due to non compliance with decision-making conditions. However, this field still poses challenges as described by the authors, and includes incorporating more complex events, analysis rules specified for particular set of level instances of a dimension level, and the fact that creation and translation of analysis rules still remains as a manual process.

Wang and Wang (2008) suggest the use of knowledge management techniques in Data Mining processes as a feasible way to capture business intelligence. Throughout a virtuous cycle, which comprise four phases: identify the problem, transform data into actionable results, act on information, and measure results, the authors suggest that business insiders can under estimate knowledge gained through Data Mining and de-emphasize roles of different people involved in the process. Therefore, a separation between business insider cycle (knowledge development) and data miner cycle (Data Mining cycle) is presented, with a common critical task of knowledge sharing and planning between them. A prototype is finally presented in this approach. Nonetheless,

the authors do not present any tools to measure its effectiveness and how it compares to the Data Mining process as noted by Kantardzic (2003).

CHAPTER III

METHODOLOGY

The experiment is divided in different aspects that will be addressed separately, which are grouped in formalism, verification and validation of the proposed method through the development of an Expert System using this technique in a practical field, specifically reprocessing in woodworking operations. Verification and validation of the proposed solution are a prerequisite for semantic validation of a knowledge base, as noted by Wentworth, Knaus and Aougab (1995).

For technique formalism, the research question of (a) how formal the new technique performs if compared with an existing one is answered. Furthermore, the new technique, basically an adaptation of commonKADS through a spiral development approach is compared with results provided by a formal commonKADS.

System verification represents one component of system testing in knowledge solutions. Similar in essence to the verification concept available in conventional software, which relates to building the system right as noted by Mosqueira-Rey and Moret-Bonillo (2000), it differs with traditional software testing since knowledge items cannot be defined precisely, correctly and completely at the beginning and may change during development. For the current study, it will address research questions related to (b) errors resulting from interactions between rules, also referred as structural errors (Huang & Cheng, 2008). Structural errors comprise redundancy, inconsistency, incompleteness and circularity of rules as noted by the authors.

Validation is the other component of knowledge system testing and also differs from conventional software as noted by Juristo and Morant (1998). It began during the development of the solution instead of an ending process of the development cycle. Nonetheless, both categories of software define validation as testing that the system output is correct and complies with needs and requirements of the user (Mosqueira-Rey & Moret-Bonillo, 2000). In the current study, it will deal with the following research questions: (c) Are meta-knowledge representations in the form of meta rules valid tools in the specification of a knowledge based solution that integrates Data Mining results in an Expert System? And (d) Can this new technique be used to create a prototype that integrates data warehousing, OLAP and Data Mining into an Expert System and compare its results with those provided by a solution obtained through a conventional method?

In order to assist the system evaluation of a practical application of the technique, the following additional research problems are also addressed: (e) Are solutions provided by this new technique effective in solving a practical domain problem? And (f) results provided by this integration are comparable with those obtained from a solution developed using a conventional approach?

The study is conducted on two main phases for implementation and evaluation of these research questions: Specification of a technique that integrates Data Mining results into an existing Expert System, and validation and verification of the expertise of the resulting knowledge system by evaluating the agreement of proposed solutions obtained from the system with the answers provided by a conventional technique.

Research Design

In evaluating formalism of a proposed technique, the Trillium scale will be used as suggested by Plant and Gamble (1997). This instrument provides an evaluation of the formal aspects that development models possess, allowing inter model comparison and establishing a ranking of formal aspects to be met by proposed models. It establishes three levels of capabilities which are cross referenced with stages of the development process, as illustrated in Table 1.

Table 1.

Trillium scale description.

	Level 1 capability	Level 2 capability	Level 3 capability
Problem specification	No explicit requirements	Informal requirements, test, plan and acceptance criteria	Semiformal statement or requirements.
Conceptual model	No documented model	Semiformal paper model. Three level component identification	Formal knowledge level model. Appropriate component representation
Design model	No design for KB. Shell used	Architectural design of components. Semiformal design of procedural parts	Formal architecture, interfaces, meta level control
Implemented model	Incomplete description	KB and inference engine traceable.	Full traceable to previous stages.
Verification analysis	Informal proofreading	KB integrity and expression logic checked. Documented anomalies	Inference logic tested. Compliance with constraints is documented
Validation analysis	Ad hoc testing. No records	Testing suite used. Semiformal usability tests	Rigorous testing with suite. Empirical methods, results fully documented.

A qualitative comparison between a combined approach of commonKADS results and the proposed technique is performed.

For verification of the resulting system, Hicks (2003) noted that not only the structure of the rule based should be tested but also each individual cluster. As indicated by Huang and Cheng (2008), structural deficiencies of a rule based solution are summarized in Table 2.

Table 2.

Structural deficiencies of rule bases.

Property	Content	Classification
Redundancy	Complete or partial redundancy in rules	Redundant rules
Inconsistency	Mutual conflict between rules	Conflicting rules
Incompleteness	Lack of rules or rules cannot be practically used	Rules with invalid conclusions or rules that do not achieve expected results
Circularity	Rule inference does not ends	Circular rules

Each aspect in the verification process of structural errors will be applied when adding new rules due to the integration of Data Mining tools. Once formally stated each entity will be proven in terms of internal redundancy, consistency, verifying that each entity is not in contradiction with other rules or leave the system in an undefined state, and for incompleteness and circularity of its structures to verify its implementation.

In order to evaluate and validate the application of the proposed technique, statistical significance of the solutions needs to be established between the Expert System obtained with the new technique and results from an Expert System obtained through conventional methods, a classical experimental design will be used for random selection of examples of rework problems on two experimental group-one control group scheme (Kerlinger, 1992). Results and solutions proposed by the Expert System and conventional method (commonKADS) will be compared using an agreement evaluation technique for expertise validation, as suggested by Mosqueira-Rey, and Moret-Bonillo (2000).

Variables and Instruments

For the formalism aspect of the proposed model, the Trillium scale is used and results are compared with analysis of another technique, commonKADS as suggested by Plant and Gamble (1997). The classification index is obtained from a direct application of the instrument, and an acceptance result is set to level 2.

For validation and verification, quantitative methods are used instead of qualitative ones. In the case of verification, acceptable requirements are identified during the development of the knowledge solution. An initial set of semantic networks are created based on the knowledge represented as rules. New rules obtained from Data Mining tools are contrasted with existing rules in terms of redundancy and conflicts through RO-RA-RV format as suggested by Huang and Cheng (2008). Every rule expressed in the form of “IF {antecedent} THEN {consequent}” is expressed in terms of relationship operator (R), object (O), attribute (A) and value (V). The relationship operator include terms $>$, $=$, $<$, $>=$ and $<=$. Logic operators AND or OR are represented as follows:

Table 3.

Logic operand equivalents in RO-RA-RV form.

Logic expression	R-O-A-V form of the antecedent
“IF (a1 AND a2) THEN c1”	[ROa1 RAa1 RVa1ROa2 RAa2 RVa2]
“IF (a3 OR a4) THEN c2”	[ROa3 RAa3 RVa3] [ROa4 RAa4 RVa4]

Results from the Expert System obtained with the proposed technique and a conventional method are tabulated using a semantic scale for symbolic classification (Semantic Scale; Mosqueira-Rey, & Moret-Bonillo, 2000) as noted in Figure 12. The experiment will require 50 cases of remanufacturing problems within different areas in the production process.

Semantic Scale
Very High (VH)
High (H)
Slightly High (SH)
Nominal (N)
Slightly Low (SL)
Low (L)
Very Low (VL)

Figure 12. Semantic Scale for agreement classification of solutions

Note: From E. Mosqueira-Rey & V. Moret-Bonillo (2000). Validation of intelligent systems: A critical study and a tool. *Expert Systems with Applications*, 18, 1-16. Copyright Elsevier Science Ltd.

Experimental Procedures and Data Analysis

In formalism evaluation, a direct evaluation of resulting values is performed compared to the acceptance level 2.

For verification purposes, each knowledge item is compared on RO-RA-RV form according with Table 4. This process will be repeated for each new rule obtained through Data Mining tools and that will be incorporated into the Expert System. By following this procedure, it ensures that new knowledge can be verified before being included in the knowledge base.

Table 4.

Redundant or conflicting rule cases.

Redundant Rules	R1: $A1 \rightarrow C1$ R2: $A1 \rightarrow C1$ Same antecedent	R1: $A1 \text{ OR } A2 \rightarrow C1$ R2: $A2 \rightarrow C1$ Same antecedent A2 and same consequent C1
	R1: $A1 \rightarrow C1 \text{ OR } C2$ R2: $A1 \rightarrow C1$ Same antecedent A1 and same consequent C1	R1: $A1 \rightarrow C1$ R2: $C1 \rightarrow C2$ R3: $A1 \rightarrow C2$ Same antecedent A1 and same consequent C2
Conflicting Rules	R1: $A1 \rightarrow C1$ R2: $A1 \rightarrow C2$ Same antecedent A1 and inconsistent consequent C1 or C2	R1: $A1 \rightarrow C1$ R2: $A1 \rightarrow \text{NOT } C1$ Same antecedent A1 and contradictory consequent C1 and NOT C1
	R1: $A1 \rightarrow C1$ R2: $A2 \rightarrow C1$ Same consequent inconsistent antecedents A1 and A2	R1: $A1 \rightarrow C1$ R2: $\text{NOT } A1 \rightarrow C1$ Same consequent C1 and conflicting antecedents A1 and NOT A1
	R1: $A1 \rightarrow C1$ R2: $A1 \rightarrow C2$ R3: $A2 \rightarrow C1$ Same antecedent A1 and consequent C1 and C2 are inconsistent in value OR same consequent C1 and inconsistent antecedents A1 and A2	

Note: From C.J. Huang & M.Y. Cheng (2008). Conflicting treatment model for certainty rule-based knowledge. *Expert Systems with Applications*, 35, 161-176. Retrieved April 1, 2011, from Elsevier Ltd Web Site: <http://www.sciencedirect.com>

Validation is evaluated by classification results of each source: The Expert System obtained through the proposed technique and the system developed through a

conventional method. Results are tabulated on a validation table as suggested by Mosqueira-Rey and Moret-Bonillo (2000).

The number of cases under each of the categories is presented on a contingency table as noted on Table 5.

Table 5.

Presentation of results and classification of agreement for each variable.

		Source 1						
		VH	H	SH	N	SL	L	VL
Source 2	VH	n_{11}	n_{12}	n_{13}	n_{14}	n_{15}	n_{16}	n_{17}
	H	n_{21}	n_{22}	n_{23}	n_{24}	n_{25}	n_{26}	n_{27}
	SH	n_{31}	n_{32}	n_{33}	n_{34}	n_{35}	n_{36}	n_{37}
	N	n_{41}	n_{42}	n_{43}	n_{44}	n_{45}	n_{46}	n_{47}
	SL	n_{51}	n_{52}	n_{53}	n_{54}	n_{55}	n_{56}	n_{57}
	L	n_{61}	n_{62}	n_{63}	n_{64}	n_{65}	n_{66}	n_{67}
	VL	n_{71}	n_{72}	n_{73}	n_{74}	n_{75}	n_{76}	n_{77}

Where n_{ij} represent the number of cases for each classification of agreement.

The dependent variable will be an agreement index calculated according to equation 1.

$$p_o = \text{Agreement Index} = \frac{\sum_{i=j}^k n_{ij}}{N} = \sum_{i=j}^k p_{ij} \quad (1)$$

Where

n_{ij} = number of cases in ij

$$N = \sum n_{ij}$$

As a measure of overall agreement between pairs, the authors suggest kappa index, defined in equation 2.

$$\text{Kappa} = \frac{p_o - p_c}{1 - p_c} \quad (2)$$

where $p_c = \sum_{i=j}^k p_{i*} p_{*j}$

The significance of the index can be tested using equation 3, by defining a level of significance of 95% ($z=1.96$; Hattie, 2001)

$$z = \frac{\text{kappa}}{\sqrt{N(1 - p_c)}} \quad (3)$$

For development purposes of the prototype, a direct application of the specification tools proposed in the new model will be used. However, in order to operate

the Trillium scale and verification purposes, each step of the development cycle will be adapted from Moreira-Rey and Moret-Bonillo (2000), as follows:

1. Requirement Analysis (RA)
2. Requirement Verification (RV)
3. Knowledge Acquisition (KA) / Data Mining - Knowledge Representation Inclusion (DMKRI)
4. Acceptance Level of Performance (ALP)
5. Prototyping
6. Verification
7. Validation

The proposed technique repeats on a similar approach as in agile development (Guckenheimer & Perez, 2006), where each new set of rules obtained through commonKADS models and Data Mining represents a new feature to be included in the iteration, until a final system is achieved. Once adopted as production system, the technique repeats only from steps 3 through 7 for newly discovered rules.

CHAPTER IV

RESULTS AND ANALYSIS OF DATA

Summary of Procedures

In order to collect data and summarize research information, the implementation of the necessary instruments to collect this data is needed. The process is divided into three main steps:

1. Starts with a knowledge based solution in the problem domain using an established technique, specifically commonKADS. Once this tool is developed,
2. A Data Warehouse that contains information regarding the problem domain: End of Line (EOL) or PONC item is implemented. It includes multidimensional objects for data presentation and analysis, and Data Mining structures and models to be implemented. These Data Mining model results are expressed in a similar knowledge representation as the system in step 1. Finally,
3. A meta rule based system is implemented using rules obtained from the Data Warehouse of PONC information. It evaluates which rules should be considered to be included on the knowledge base of the initial solution for PONC analysis.

Step 1. Development of a knowledge based system using commonKADS

This development process is based on Moreira-Rey and Moret-Bonillo (2000) who suggested a spiral methodology for intelligent systems based on Lee and O'keefe model.

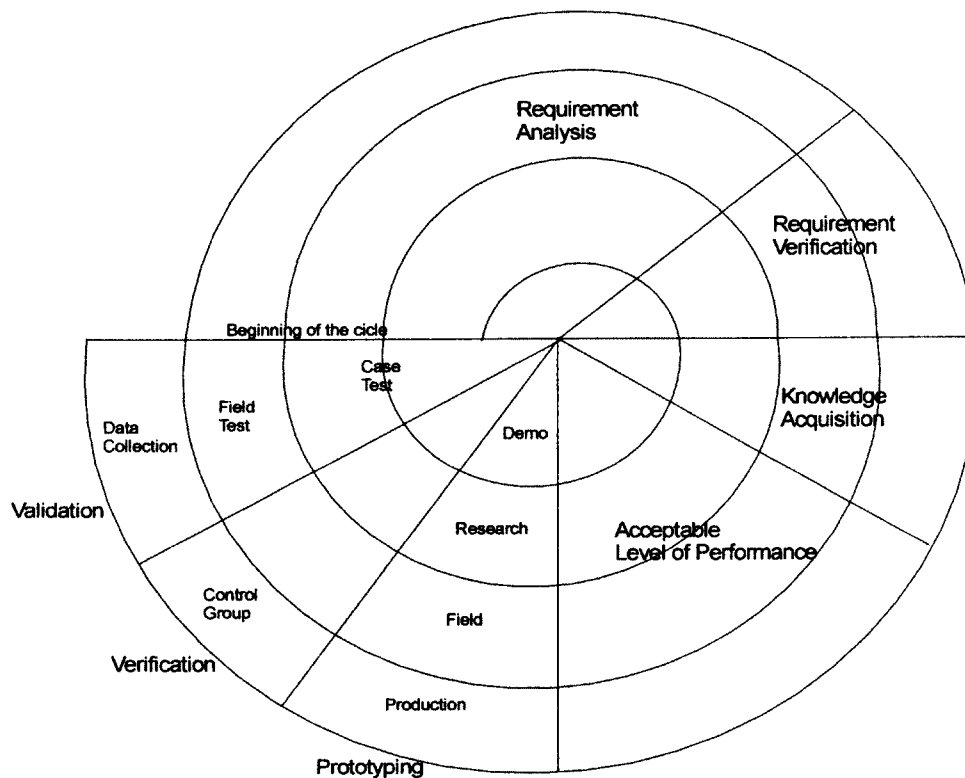


Figure 13. Spiral development approach.

Note: Adapted from Mosqueira-Rey and Moret-Bonillo (2000). Validation of intelligent systems: a critical study and a tool. *Expert Systems with Applications*, 18. 1-16.

This spiral approach together with the formal tools provided by CommonKADS for the stages of Requirement Analysis, Requirement Verification, Knowledge Acquisition and Acceptance Levels of Performance ensure correctness and completeness of the solution. A simplified approach of CommonKADS is used for all models created by this methodology, as presented in Figure 14. An emphasis is placed in the design model and knowledge model since they are the key elements for prototyping, verification and validation stages.

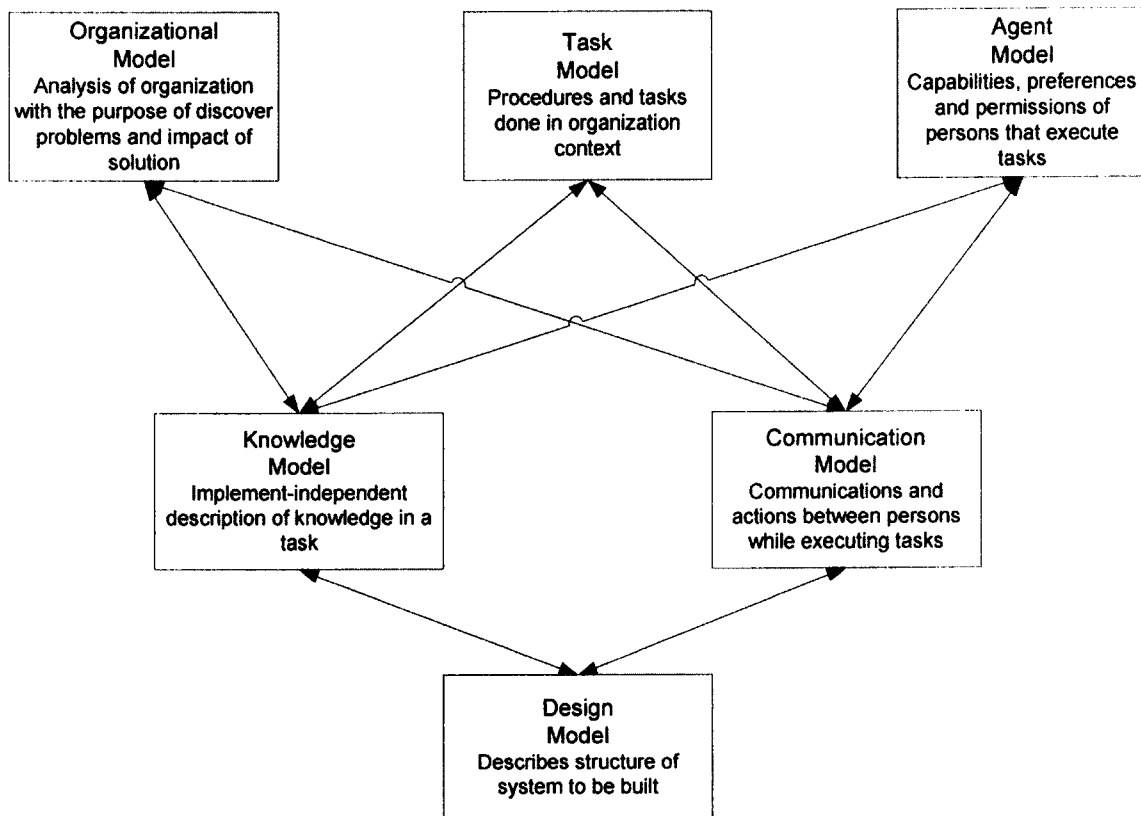


Figure 14. CommonKADS methodology models

Note: Adapted from Akkermans et al (1993). *Expertise model definition document*

(Deliverable ESPRIT Nr. P5248 KADS-II/M2/UvA/0026/1.1. Rapport aan: EC.).

Retrieved March 2004 from <http://swi.psy.uva.nl/projects/CommonKADS/postscript/CK-UvA-26.ps.gz>.

The Knowledge Model construction comprise three stages as noted by Schreiber (2010): (a) knowledge identification, which provides an initial familiarization with the domain and potential components for reuse, (b) knowledge specification, with a task template selection and a complete knowledge model, and (c) knowledge refinement, where an initial prototype is created and additional knowledge is incorporated.

To complete knowledge identification (a), it is required to determine the problem domain of the Expert System to be developed. The problem domain selected is cabinet manufacturing reordering for remake and rework. The operations included for remake/rework comprise order intake, product design, component creation, lumber and millwork for woods, veneer, plastic and glass, sawing, surfacing, drilling and boring, shaping, turning, assembly and final product handling and distribution all main components of the cabinetmaking process as noted by Umstattd and Davis (2000). The scope of operations extends to 5 lines ranging from fully custom to standard or stock cabinets, bath line, glass, doors, and cabinet tops. The location of the problem domain is a private local company that uses the methodology of Price of Non-Conformance (PONC) as the value for quality problems. This methodology, as indicated by Crosby (2004) specified that:

1. All work is a process,
2. The definition of quality is conformance to requirements, and
3. The price of non-conforming to the customer requirements is the cost of quality by which it can be measured.
4. This method implies that the standard expected in quality is zero defects.

Consequently, the company implemented a tracking system to identify and record the cost of non-conforming or PONC.

Although this method sets an agreement on how to measure quality, there are still differences on how to account for those opportunities internally. If a defective piece is received, it is important not only to know the value of that defective part, but also where

it needs to be addressed so it will not happen again. This latter question requires a more in-process analysis and diagnostic that is not structured and depends on knowledge of business analysts that evaluate on different areas what needs to be done and who needs to be acquainted to implement a fix or a solution when needed.

To identify the knowledge to be included (b), an analysis of PONC assigned to Research and Development (RND), Information Systems (IS), and Data Distribution Group (DDG) in the last 3 months is performed. This analysis comprises attributes related to the PONC reordering, order details, modifications applied to the order, production order (PO), notifications sent about changes in the order or PO, formulas used to obtain components, assembly sheets used to build the cabinet or cut list reports employed to dimension material. The relationship between these objects used to determine PONC knowledge is illustrated in Figure 15.

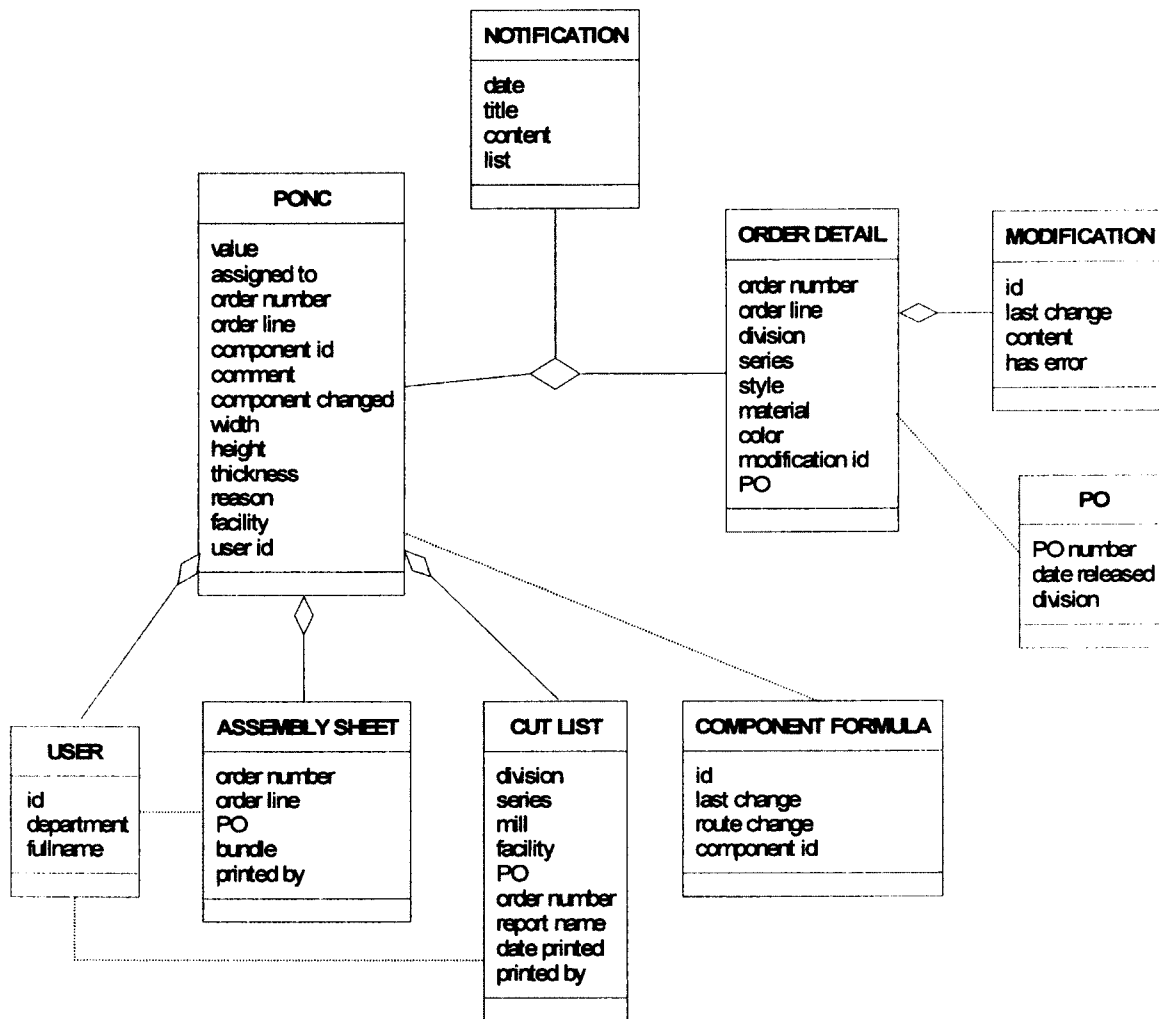


Figure 15. PONG related objects and attributes used in knowledge identification

With these objects identified, knowledge refinement (c) was used to obtain a basic set of rules that covered most of the cases analyzed by RND, IS and DDG for the time period specified of 3 months. This refinement produced 7 rules with 2 actions per rule based on PONG assigned department, as noted in Table 6.

Table 6.

Rules obtained after knowledge refinement.

Rule	Action
R1. PONC reason is "NOT ON ASSEMBLY SHEET" AND PONC component has changed → "Component was changed but not included initially on assembly sheets"	PONC assigned to DDG → keep PONC not assigned to DDG → reassign to DDG
R2. PONC reason is "NOT ON CUT LIST" AND PONC component has changed AND Order detail division is "CUSTOM" AND order detail reprocess date is greater than date PO was released → Changes were made to the order after it was released for production	PONC assigned to CDS → keep PONC not assigned to CDS → reassign to CDS
R3. PONC reason is "NOT ON CUT LIST" AND PONC component has changed AND Order detail division is "BATH" AND Order detail reprocess date is greater than date PO was released → Item was scheduled in BDS and then rescheduled after paperwork was printed	PONC assigned to BDS → keep PONC not assigned to BDS → reassign to BDS
R4. PONC reason is "NOT ON CUT LIST" AND PONC component has changed AND notification date is greater or equal to date cut list was printed → Order may have been reprocessed and changes were not reflected on paperwork	PONC assigned to DDG → keep PONC not assigned to DDG → reassign to DDG
R5. PONC reason is "CUT LIST WRONG" AND PONC component has not changed AND last change of modification > date PO was released → Modifications on the order were changed after the component was scheduled on PO	PONC assigned to RND → keep PONC not assigned to RND → reassign to RND
R6. PONC reason is "CUT LIST WRONG" AND PONC component has changed AND last change of modification < date PO was released → Modification needs to be updated	PONC assigned to RND → keep PONC not assigned to RND → reassign to RND
R7. PONC reason is "LIST ERROR" OR PONC reason is "NOT ON ASSEMBLY SHEET" AND order detail catalog contains "SHIP-LOOSE FOR" AND PONC comment contains "QUANTITY" AND PONC component was not changed → Potential system issue with ship loose items quantity	PONC assigned to IS → keep PONC not assigned to IS → reassign to IS

For the Design model and implementation of the knowledge based solution, a quick development approach based on agile principles was used following the last three

stages of the spiral approach of Mosqueira-Rey and Moret-Bonillo (2000). This approach comprised knowledge representation structures summarized in a rule set. The implementation tool used was Visual Rule Studio 2.5 which embedded rules inside a standard visual basic programming environment for windows operating system. The program code structure produced after several iterations for PONC Expert System (PONC Expert) is noted in Figure 16.

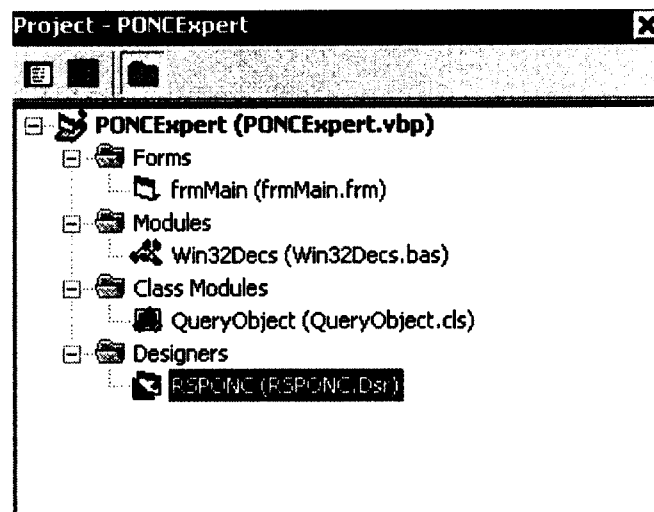


Figure 16. Project PONC Expert code structure

An emphasis was placed on a simplified interface and flexibility on the user entry screen as noted in Figure 17. Two main areas are available for the user to interact, a Question part where inquires in plain English are stated, and a variable object area underneath where the user must enter a response.

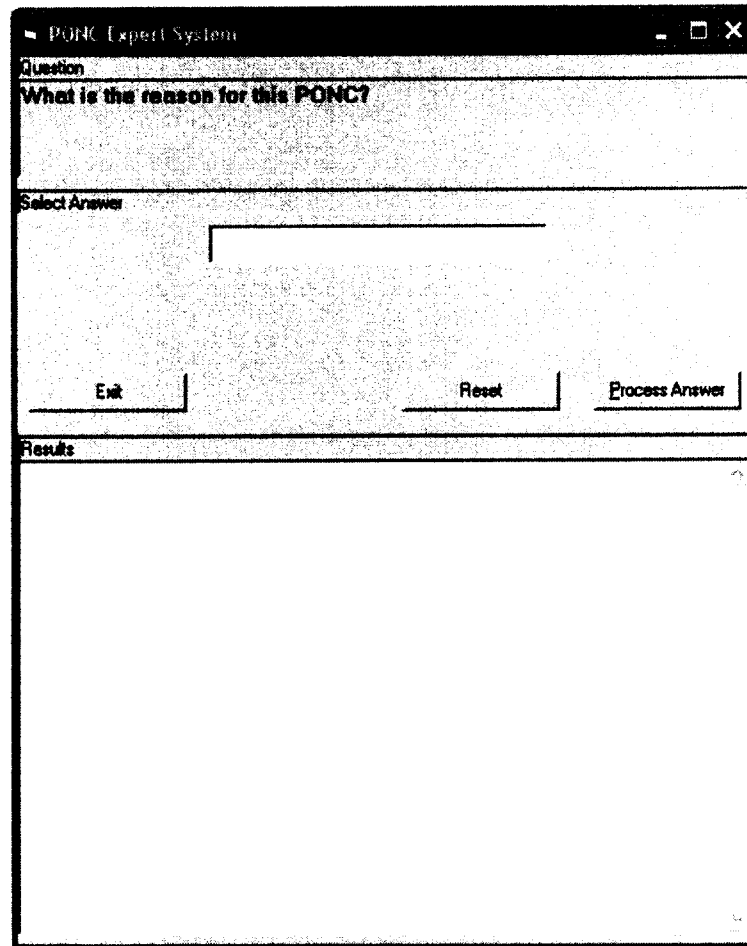


Figure 17. General PONC Expert interface

The Select Answer area may contain selection lists, text boxes or true/false buttons depending on the expected answer type as noted in Figure 18.

The user must press the process answer button to continue to the next question. The knowledge system uses backward chaining as inference engine mechanism at the moment, and the language allows for easily set up of an agenda of goals to be achieved.

If a question is left unanswered, the Expert System assumes an unknown state where conclusions may not be achieved. The development tool, Visual Rule Studio,

provides a mechanism to allow inference even with unanswered questions based on a confidence factor and a rule set threshold, which is set to 50 by default.

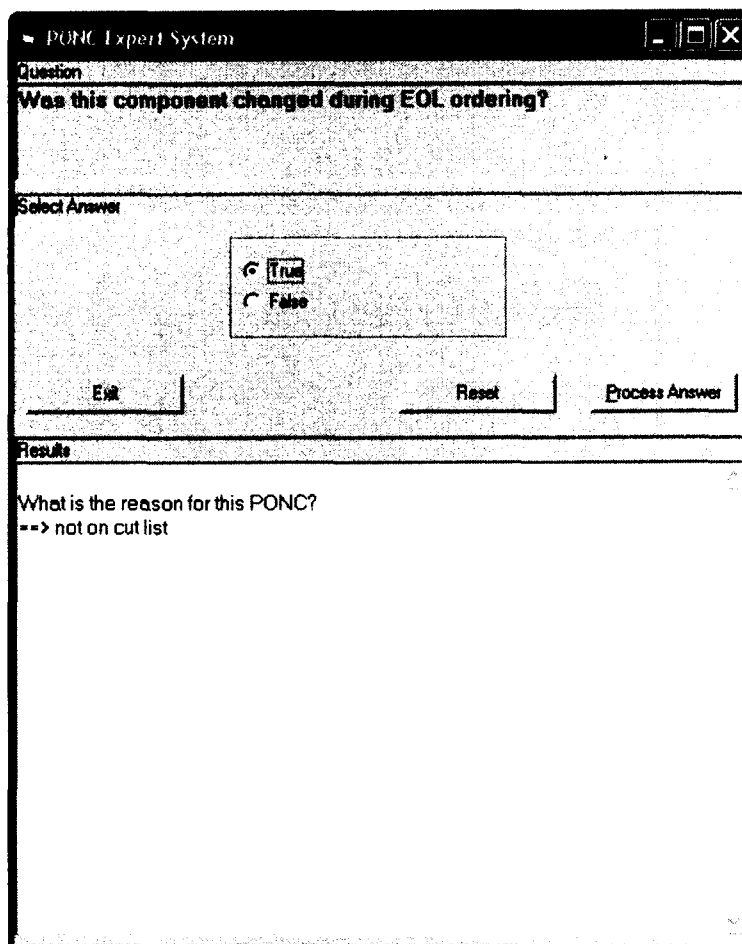


Figure 18. PONC Expert with a true/false question object in answer window

The bottom part of the screen summarized the interaction for later review as noted in Figure 19. It includes all questions and answers for the current session as well as conclusions and recommendations obtained by the Expert System. The user can restart

the session at any time using the reset button, clearing out the answers and restarting questions from the beginning.

The screenshot shows a window titled "PONC Expert System" with a standard Windows-style title bar. The interface is divided into several sections:

- Question:** "Enter date when order was release to production"
- Select Answer:** A text input field containing "11/25/2011".
- Buttons:** "Exit", "Reset", and "Process Answer".
- Results:** A scrollable area containing the following text:
 - What is the reason for this PONC? --> not on cut list
 - Was this component changed during EOL ordering? --> True
 - What department is this PONC currently assigned? --> ddg
 - What is the order division? --> custom
 - When was this order reprocessed? --> 11/26/2011
 - Enter date when order was release to production --> 11/25/2011
 - Explanation: Changes were made to the order after it was released for production
 - Instructions: reassign_to_CDS

Figure 19. PONC Expert Results screen region with complete session interaction

An initial model of the rule set is presented in Figure 20. Attributes labeled on a circle are input and those labeled with circle and a centered dot are conclusion or potential agenda goals, as in suggested_action and explanation in EOL object.

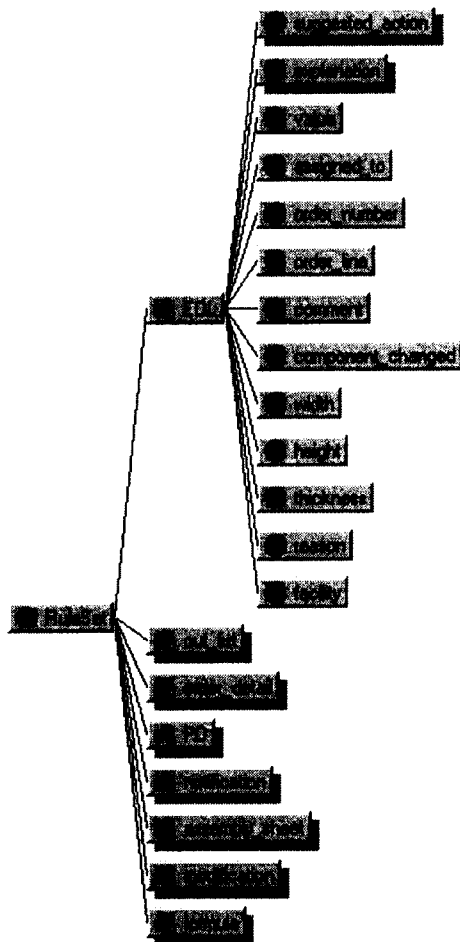


Figure 20. RuleSet obtained through knowledge refinement

Step 2. Data Warehouse implementation for EOL PONC information

Using embedded functionality on existing database manager software, a complete Data Warehouse was implemented for EOL PONC information using MS SQL 2005 Analysis Services. A prior step was importing into a normalized MS SQL version of all data related to EOL PONCs for 3 months. This prior step required all dependencies being migrated from production live data. Although not a necessary step, it was performed to ensure complete integration between different steps in the Data Warehouse implementation. First, dimensions were identified and measurements were defined as illustrated in Figure 21. Once dimension and measurements were created, a cube was implemented to allow slice and dice functionality for additional analysis. All information summarized in the cube is connected to the database through a data source. This element of the Data Warehouse represents the connection to live data. Once recognized, a data source view is required to set up a relationship among fields to be included in the cube, as noted in Figure 22. This data source view is also used during Data Mining construction of their structure and model selected. The cube is defined by dynamically specifying a measurement and dimensions to be included. Nonetheless, not all dimensions are related to each measurement since they also obey to the relationships indicated in the data source view. The capability of dissecting information based on dimensions is the key feature that cubes provide for effective decision making, as suggested by Lavene and Loizou (2003). Furthermore, the authors suggest that an effective representation of a Data Warehouse should be similar to the snowflake design, specifically the star scheme.

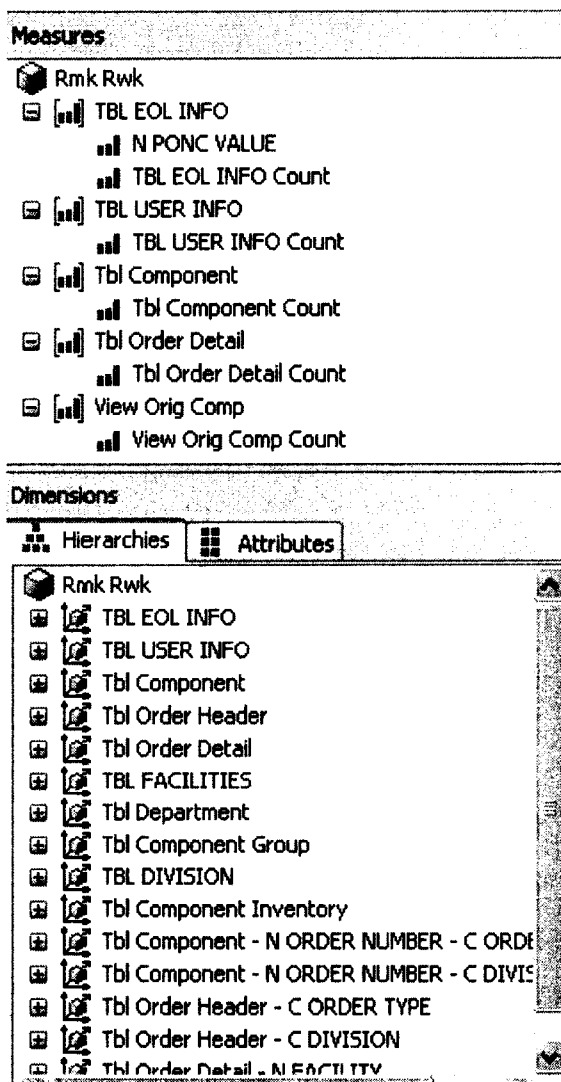


Figure 21. Dimensions and measurements defined for EOL PONC Data Warehouse

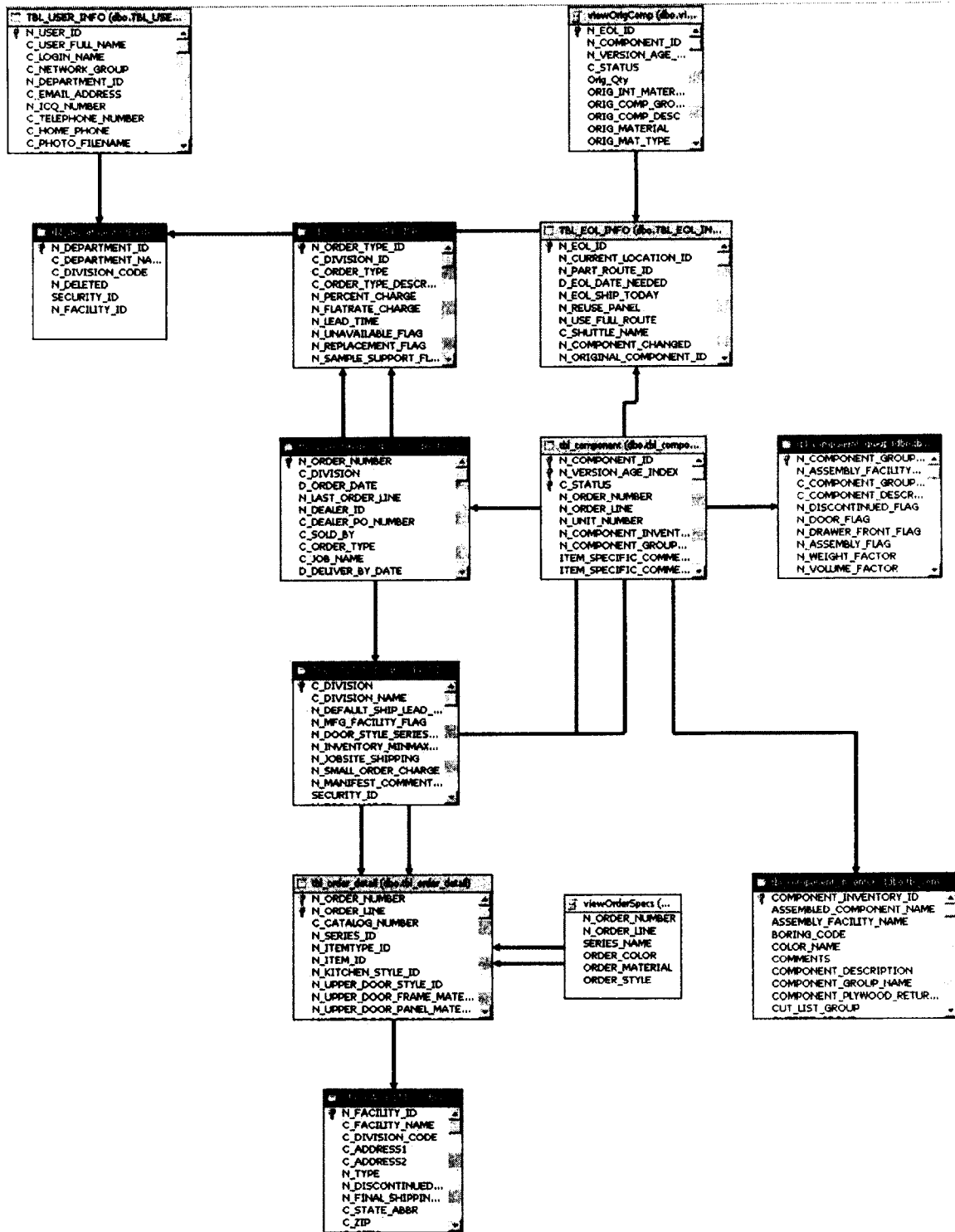


Figure 22. Data source view model for EOL PONC Data Warehouse cube definition and Data Mining

An example of one dimension and measurement interaction is presented in Figure 23. A detail of each department cost in terms of PONC is presented. However, division can also be included as an aggregation dimension, resulting in a more complex cube. This dimension aggregation can also be restrained by selecting a specific dimension as noted in Figure 24. This technique consists on selecting a specific value for the selected dimension and project results over the rest dimensions.

DEPARTMENT	PONC VALUE
080	\$2,872.34
060	\$1,292.95
100	\$666.95
120	\$1,009.98
070	\$2,062.37
090	\$714.38
071	\$356.74
010	\$1,638.86
020	\$1,694.96
030	\$19,693.79
040	\$1,722.06
050	\$198.44
060	\$4,436.43
070	\$4,364.22
080	\$13,724.64
090	\$919.15
100	\$217.32
110	\$3,529.83
120	\$1,366.74
130	\$282.21
140	\$726.92
150	\$24.18
160	\$435.38
170	\$1,718.71
180	\$544.66
190	\$565.53
200	\$497.33
210	\$2,985.35
220	\$1,221.32
230	\$887.46

Figure 23. One dimension and one measurement for multidimensional analysis (cubes)

COMPONENT GROUP NAME		C DEPARTMENT NAME									Grand Total
APPLIED MOLDING		R60	R40	L50	L70	M50	PURCHASING	R30	RND		
TYPE NAME	N PONC VALUE	N PONC VALUE	N PONC VALUE	N PONC VALUE	N PONC VALUE	N PONC VALUE	N PONC VALUE	N PONC VALUE	N PONC VALUE	N PONC VALUE	
CAPELA MOLDING				\$19.00	\$1.68	\$0.68	\$20.58	\$32.65	\$0.00	\$32.65	
CRAFTWOOD MOLDING								\$2.71		\$44.65	
TIFFANY MOLDING	\$12.65	\$32.26	\$0.41					\$28.13		\$73.45	
Grand Total	\$12.65	\$32.26	\$0.41	\$19.00	\$1.68	\$0.68	\$20.58	\$63.49	\$0.00	\$150.75	

Figure 24. Multidimensional cube with dimension projection for EOL PONC

Data Mining in Analysis services allows developers to implement a mining structure which can contain several models, as indicated by SQL Server 2005 Books Online (2008). This process of obtaining valid information from large databases as noted by Larose (2006) was defined with a simple structure of one view. On this structure, a model was selected among the following options for Data Mining EOL PONC information:

1. Classification algorithms. Predict one or more variables based on other fields in the data set. Variables are discrete
2. Regression algorithms, which predict variables based on continuous attributes on a data set.
3. Segmentation algorithms, dividing data into clusters of items with similar characteristics.
4. Association rules algorithms, which identifies correlations between different attributes on a data set. It identifies items that usually happen in sequence (Albion Research Ltd., 2011)
5. Sequence analysis algorithms, which summarize sequence of data or series.

The selected algorithm to determine which attributes are more likely to appear together for a PONC was Association rules algorithm. This algorithm represented a direct

implementation of the Apriori algorithm, which does not analyze patterns but instead generates all possible combinations of item sets. For each item set, the algorithm calculates the number of cases that contain the item set and their threshold provides with two major values for each combination of item sets or possible candidate attributes to be included in the analysis: (a) probability for a rule to be true, and (b) importance which is a factor that measures how much is more likely for Right Hand Side (RHS) to appear together with Left hand Side (LHS) than without. A decision tree algorithm is suggested to be used to estimate PONC values, however further analysis is not included in the current study.

Data Mining models represent an implementation of a learning machine, as described by Kantardzic (2003). These machines are a combination of artificial intelligence and statistics with the most common task being inductive machine learning according to the author. A general overview of a learning machine is presented in Figure 25. The learning machine creates an approximated response as outputs from observations of the system.

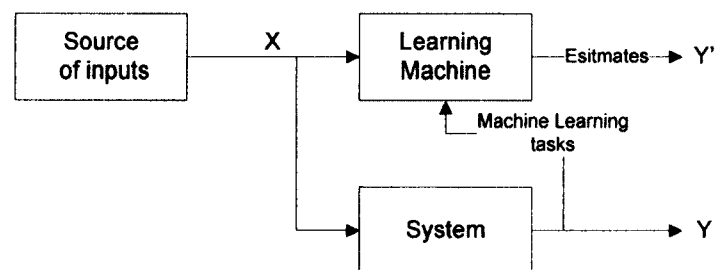


Figure 25. Learning machine view.

Note: Adapted from Kantardzic. (2003). *Data Mining. Concepts, models, methods, and algorithms*. Piscataway, NJ: IEEE Press.

The selected algorithm used a training data set of 8,674 EOL PONC cases for a time period of August 1, 2011 until November 26, 2011. Concurrently with results obtained from Microsoft Analysis Services 2005, an additional analysis of resulting rules was performed in Clementine 7.0 software to validate mining results on the selected training data set as noted in Figure 26 and 27.

Structure	PONC Department Association M...	PONC Value Decision Model	PONC Value Association Model	PONC Department Decision Model
	Microsoft_Association_Rules	Microsoft_Decision_Trees	Microsoft_Association_Rules	Microsoft_Decision_Trees
C DIVISION NAME	Input	Input	Input	Input
C LOCATION NAME	Input	Input	Input	Input
C ORDER TYPE DESCRIPTION	Input	Input	Input	Input
C REASON	PredictOnly	PredictOnly	PredictOnly	PredictOnly
COLOR NAME	Input	Input	Input	Input
CUT LIST GROUP	Input	Input	Input	Input
D DATE TIME	Input	Input	Input	Input
FACILITY NAME	Input	Input	Input	Input
GLAZE	Input	Input	Input	Input
INTERIOR MATERIAL NAME	Input	Input	Input	Input
MATERIAL NAME	Input	Input	Input	Input
MATERIAL TYPE NAME	Input	Input	Input	Input
N COMPONENT QUANTITY	Input	Input	Input	Input
N EOL ID	Key	Key	Key	Key
N PONC VALLE	Ignore	PredictOnly	PredictOnly	Ignore
N PRODUCTION WEEK	Input	Input	Input	Input
PONC DEPARTMENT	PredictOnly	Ignore	Ignore	PredictOnly
PRODUCTION GROUP	Input	Input	Input	Input
STYLE NAME	Input	Input	Input	Input
SUBASSEMBLY FLAG	Input	Input	Input	Input
THICKNESS	Input	Input	Input	Input

Figure 26. Data Mining models in Analysis Services 2005 for department and value rules

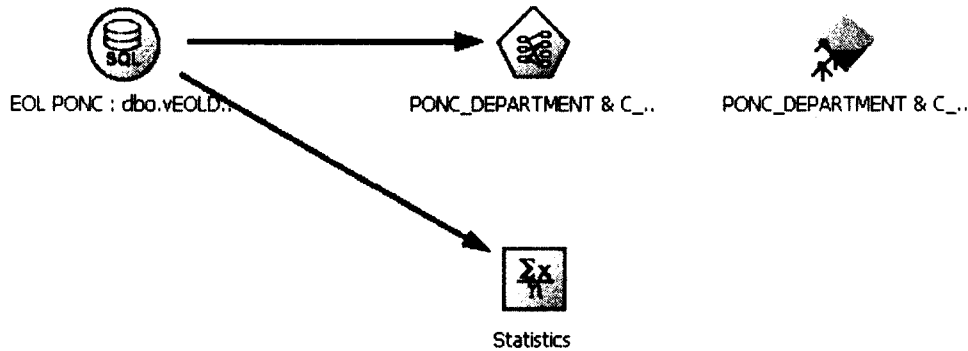


Figure 27. Clementine model for PONC department estimation using association rules

These models identified 156 candidate rules for departments D-21 (Rough Mill) and F-40 (Finish Mill). The parameters used in each program are equivalent and they created the same rules set, with a difference in Clementine that considered only 153 potential rules. For Analysis Services, a base importance of 0.27 and minimum probability of 0.4 was used. In Clementine, minimum rule support was set to 13.5 and used the same value of 40% as rule confidence. Both programs execute the Apriori algorithm, which computes how frequent sets of attributes are on a two step process: (a) Attribute set generation and (b) set counting and selection based on criteria values as described by Kantardzic (2003). The tabular representation of each program is presented in Figure 28 and Figure 29.

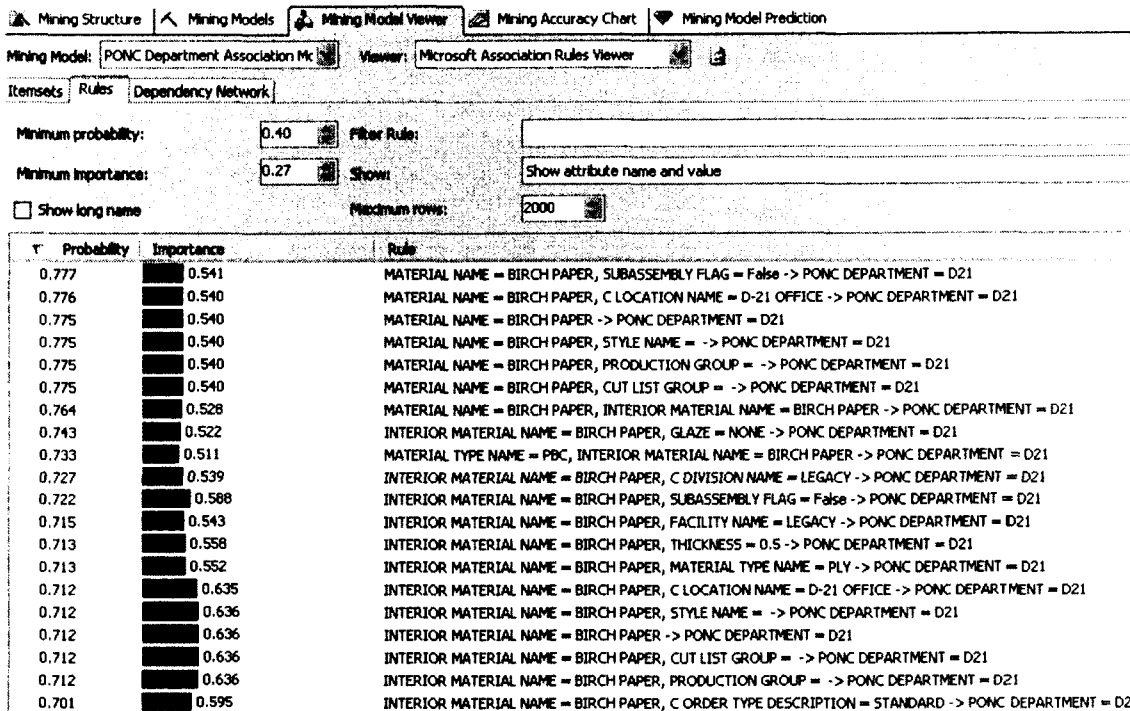


Figure 28. Rule selection for item sets defined in Analysis Services 2005 for EOL PONC in tabular form.

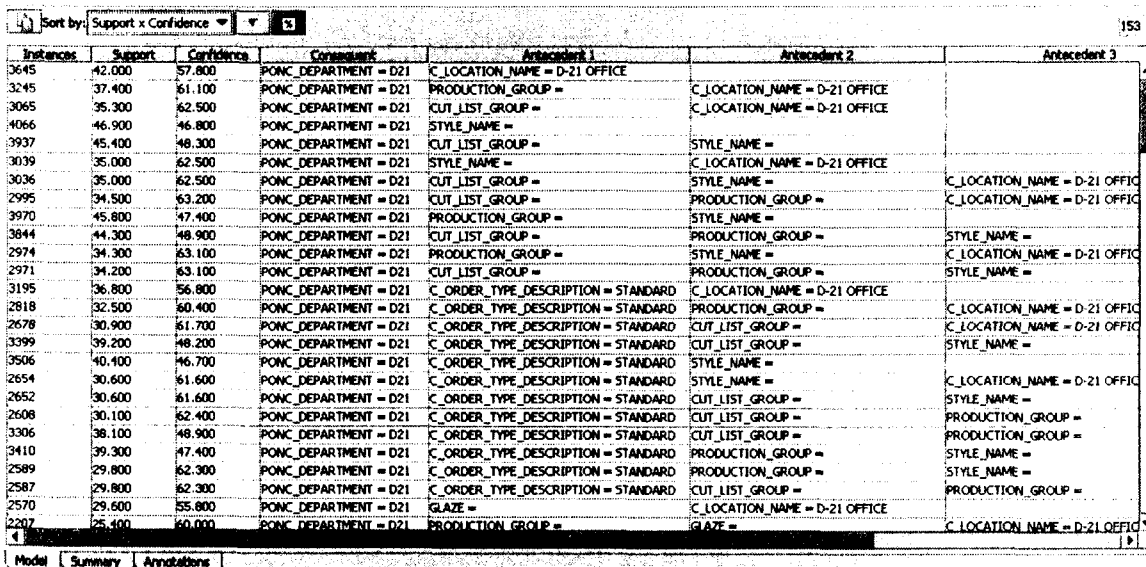


Figure 29. Candidate rules table obtained in Clementine with correspondent item sets for EOL PONC cases

Furthermore, Clementine allowed visualization of candidate rules in a user friendly manner as presented in Figure 30.

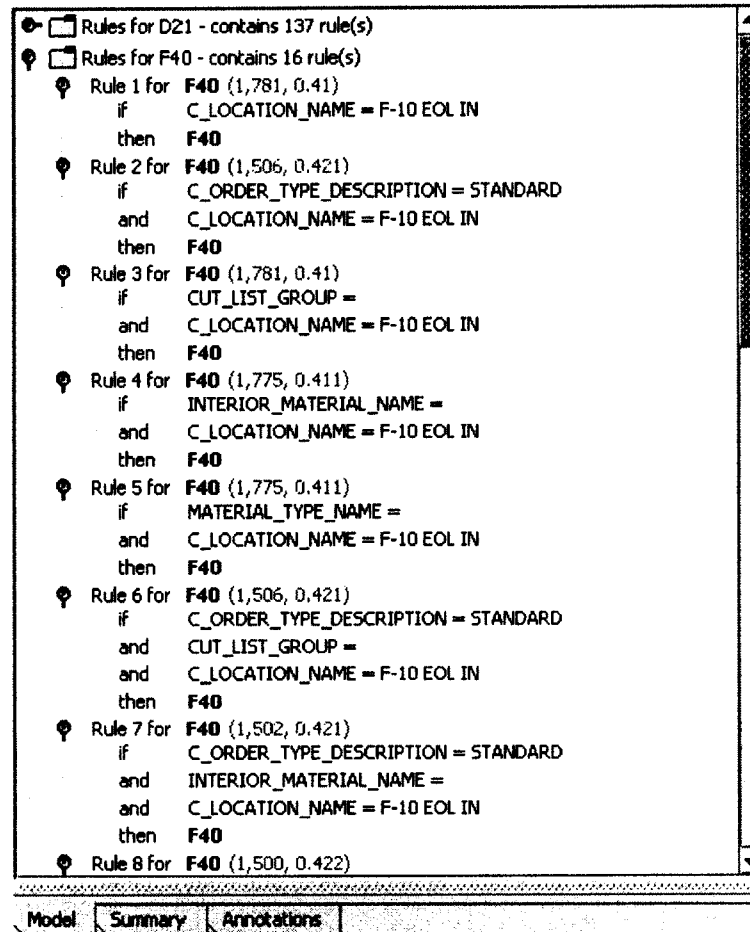


Figure 30. Candidate rules obtained in Clementine.

A final step was required to review results and save these candidate rules in a format that meta rules can be applied to it directly considering that Analysis Services 2005 do not provide this functionality for each end user. For this purpose, a VB .NET 2005 program was created to visualize Data Mining models stored on a Data Warehouse

and an option to save these results as an input file for next processing step as presented in Figure 31 and Figure 32.

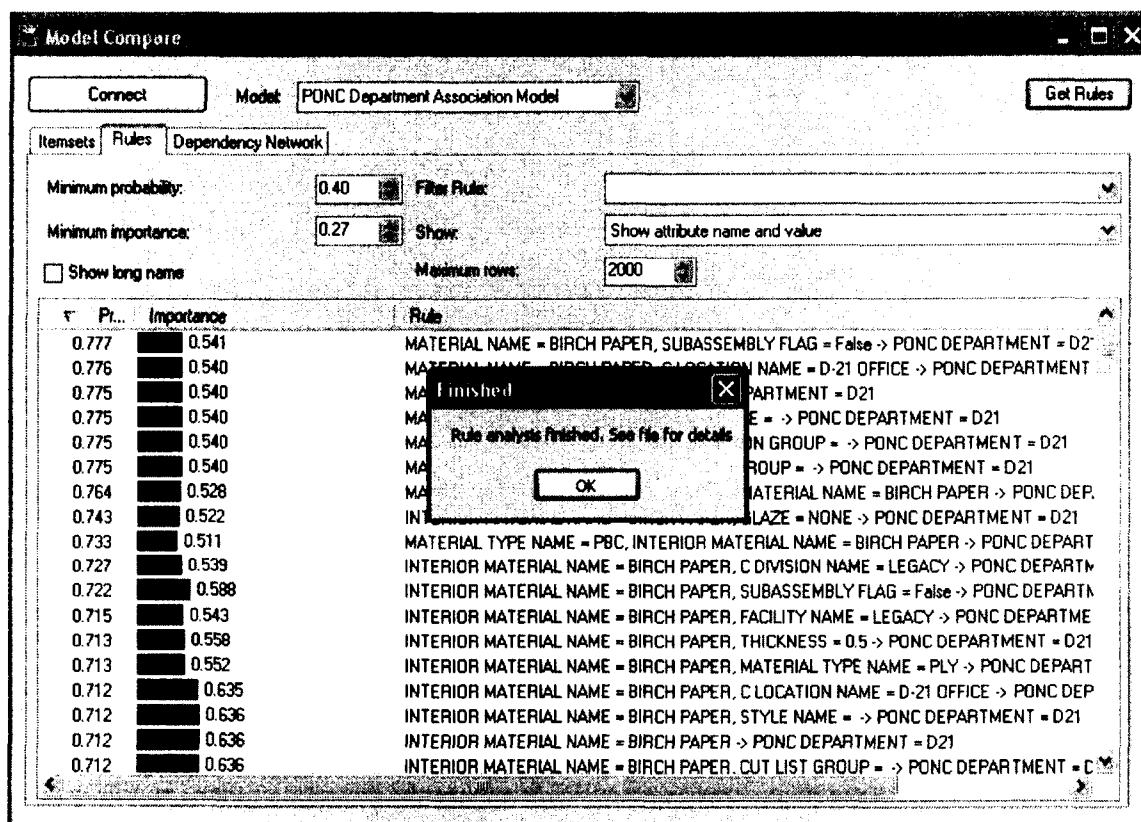


Figure 31. VB .NET 2005 program used to save obtained rules from Association rules method in a file for further processing

```

DMMRules.txt Notepad
File Edit Format View Help
MATERIAL NAME = BIRCH PAPER, SUBASSEMBLY FLAG = Missing -> PONC DEPARTMENT = D21
MATERIAL NAME = BIRCH PAPER, C LOCATION NAME = D-21 OFFICE -> PONC DEPARTMENT = D21
MATERIAL NAME = BIRCH PAPER -> PONC DEPARTMENT = D21
MATERIAL NAME = BIRCH PAPER, INTERIOR MATERIAL NAME = BIRCH PAPER -> PONC DEPARTMENT = D21
INTERIOR MATERIAL NAME = BIRCH PAPER, GLAZE = NONE -> PONC DEPARTMENT = D21
MATERIAL TYPE NAME = PBC, INTERIOR MATERIAL NAME = BIRCH PAPER -> PONC DEPARTMENT = D21
INTERIOR MATERIAL NAME = BIRCH PAPER, C DIVISION NAME = LEGACY -> PONC DEPARTMENT = D21
INTERIOR MATERIAL NAME = BIRCH PAPER, SUBASSEMBLY FLAG = Missing -> PONC DEPARTMENT = D21
INTERIOR MATERIAL NAME = BIRCH PAPER, FACILITY NAME = LEGACY -> PONC DEPARTMENT = D21
INTERIOR MATERIAL NAME = BIRCH PAPER, THICKNESS = 0.5 -> PONC DEPARTMENT = D21
INTERIOR MATERIAL NAME = BIRCH PAPER, MATERIAL TYPE NAME = PLY -> PONC DEPARTMENT = D21
INTERIOR MATERIAL NAME = BIRCH PAPER, C LOCATION NAME = D-21 OFFICE -> PONC DEPARTMENT = D21
INTERIOR MATERIAL NAME = BIRCH PAPER -> PONC DEPARTMENT = D21
INTERIOR MATERIAL NAME = BIRCH PAPER, C ORDER TYPE DESCRIPTION = STANDARD -> PONC DEPARTMENT = D21
INTERIOR MATERIAL NAME = BIRCH PAPER, N COMPONENT QUANTITY = 1 -> PONC DEPARTMENT = D21
THICKNESS = 0.5, GLAZE = NONE -> PONC DEPARTMENT = D21
INTERIOR MATERIAL NAME = BIRCH PAPER, C DIVISION NAME = BATH -> PONC DEPARTMENT = D21
MATERIAL TYPE NAME = PLY, FACILITY NAME = BATH -> PONC DEPARTMENT = D21
THICKNESS = 0.5, MATERIAL TYPE NAME = PLY -> PONC DEPARTMENT = D21
THICKNESS = 0.5, FACILITY NAME = LEGACY -> PONC DEPARTMENT = D21
SUBASSEMBLY FLAG = Existing, C LOCATION NAME = D-21 OFFICE -> PONC DEPARTMENT = D21
MATERIAL TYPE NAME = PLY, SUBASSEMBLY FLAG = Existing -> PONC DEPARTMENT = D21
THICKNESS = 0.5, C DIVISION NAME = BATH -> PONC DEPARTMENT = D21
THICKNESS = 0.5, C DIVISION NAME = LEGACY -> PONC DEPARTMENT = D21
FACILITY NAME = BATH, C LOCATION NAME = D-21 OFFICE -> PONC DEPARTMENT = D21
THICKNESS = 0.5, C LOCATION NAME = D-21 OFFICE -> PONC DEPARTMENT = D21
THICKNESS = 0.5, N COMPONENT QUANTITY = 1 -> PONC DEPARTMENT = D21
INTERIOR MATERIAL NAME = BIRCH, FACILITY NAME = BATH -> PONC DEPARTMENT = D21
THICKNESS = 0.5 -> PONC DEPARTMENT = D21
INTERIOR MATERIAL NAME = BIRCH, SUBASSEMBLY FLAG = Existing -> PONC DEPARTMENT = D21
MATERIAL TYPE NAME = PLY, GLAZE = NONE -> PONC DEPARTMENT = D21
N PRODUCTION WEEK = 40, C LOCATION NAME = D-21 OFFICE -> PONC DEPARTMENT = D21
THICKNESS = 0.5, SUBASSEMBLY FLAG = Missing -> PONC DEPARTMENT = D21
MATERIAL TYPE NAME = PLY, C DIVISION NAME = LEGACY -> PONC DEPARTMENT = D21
THICKNESS = 0.5, C ORDER TYPE DESCRIPTION = STANDARD -> PONC DEPARTMENT = D21
GLAZE = NONE, C LOCATION NAME = D-21 OFFICE -> PONC DEPARTMENT = D21
MATERIAL TYPE NAME = PBC, C LOCATION NAME = D-21 OFFICE -> PONC DEPARTMENT = D21
INTERIOR MATERIAL NAME = BIRCH, C DIVISION NAME = BATH -> PONC DEPARTMENT = D21
MATERIAL TYPE NAME = PBC, THICKNESS = 0.5 -> PONC DEPARTMENT = D21
MATERIAL TYPE NAME = PLY, THICKNESS = 0.75 -> PONC DEPARTMENT = D21

```

Figure 32. Data Mining candidate rules exported to file.

Step 3. Meta rule system development for analysis of rules obtained from Analysis Services

Using meta rules, rules on how to integrate new rules obtained from Data Mining and perform simple rule verification, the development of a new knowledge based system is performed. MetaRules program is created using Visual Basic 6 and Visual Rule Studio 2.5, as host of the automated set of meta rules as illustrated in Figure 33. This program uses forward chaining as inference mechanism and reads directly from file created with candidate rules.

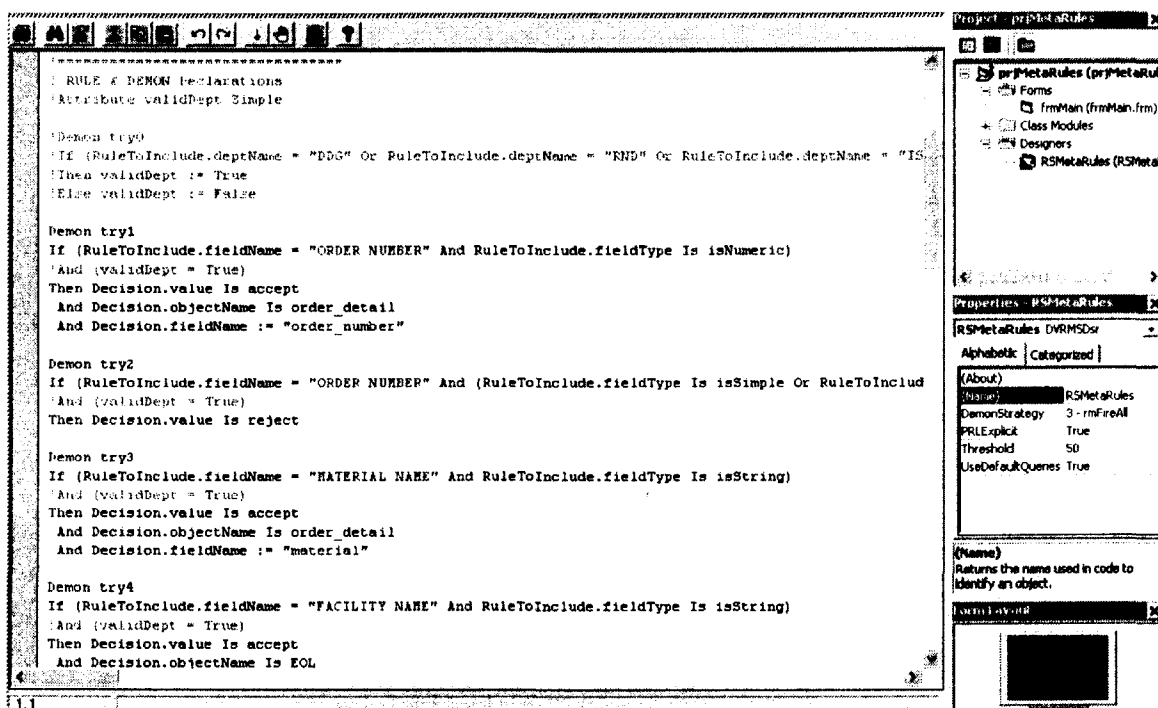


Figure 33. Meta rules embedded in automated program used to create final rules to be used in PONC Expert

The resulting set of rules in standard PONC nomenclature of fields and logic is saved as a new file as noted in Figure 34 and Figure 35. Metarules comprise a series of rules which verify that candidate rules have a complete form with all fields not considering incomplete conditions or null values, and convert each selected rule into a new syntax that can be directly used in the PONC analysis Expert System.

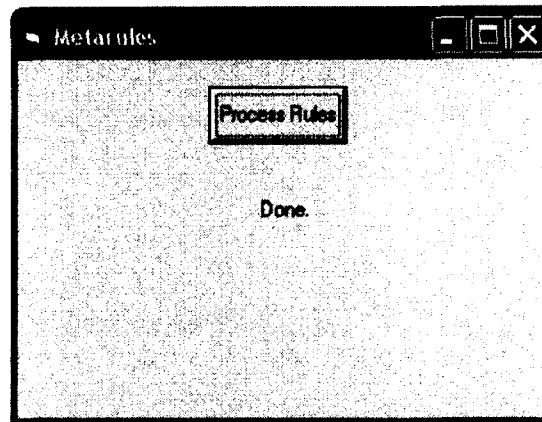
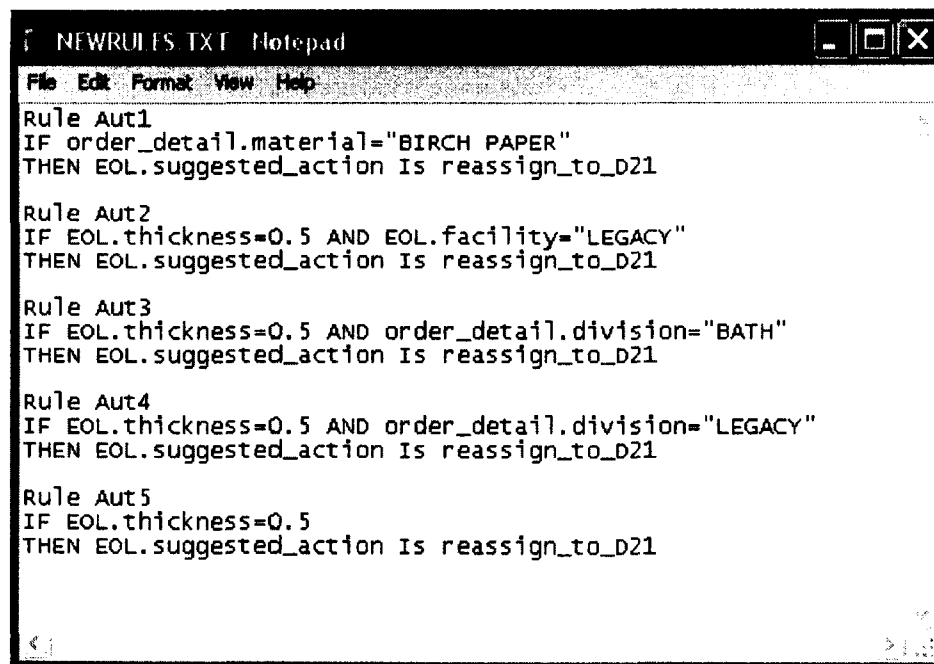


Figure 34. Metarules processing screen



```
File Edit Format View Help
Rule Aut1
IF order_detail.material="BIRCH PAPER"
THEN EOL.suggested_action Is reassign_to_D21

Rule Aut2
IF EOL.thickness=0.5 AND EOL.facility="LEGACY"
THEN EOL.suggested_action Is reassign_to_D21

Rule Aut3
IF EOL.thickness=0.5 AND order_detail.division="BATH"
THEN EOL.suggested_action Is reassign_to_D21

Rule Aut4
IF EOL.thickness=0.5 AND order_detail.division="LEGACY"
THEN EOL.suggested_action Is reassign_to_D21

Rule Aut5
IF EOL.thickness=0.5
THEN EOL.suggested_action Is reassign_to_D21
```

Figure 35. Rules obtained after meta rules analysis in a standard form to be included in PONC Expert System

These knowledge based systems and tools will be used to collect data and respond research questions as noted earlier in the study:

1. How formal this new technique performs if compared with an existing one such as pure commonKADS?
2. Does this technique address errors resulting from interactions between rules, also referred as structural errors?
3. Can meta rules be used in the integration of new knowledge obtained through Data Mining methods into a knowledge based solution?
4. Can this new technique be used to create a working knowledge system that integrates data warehousing, OLAP and Data Mining with an Expert System? Correspondingly, are solutions provided by this new technique effective in solving a practical domain problem?
5. Results provided by this integration are comparable with those obtained from a solution developed using a conventional approach?

CHAPTER V

SUMMARY, CONCLUSIONS AND RECOMMENDATIONS

Report of Results

In order to respond to research question 1, a Trillium scale proposed by Plant and Gamble (1997) was evaluated using the modified technique that combine commonKADS principles, centered on Knowledge model and Design model, and spiral development suggested by Mosqueira-Rey and Moret-Bonillo (2000) as illustrated in Figure 36. This technique entails continuous integration of new rules or knowledge representation forms to the developed system on each stage of overall development. In the same way, the technique iterates following analogous principles as in agile development, adding planned features to the developed system in a given iteration. Using this technique, a new knowledge based system was implemented, named PONC Expert.

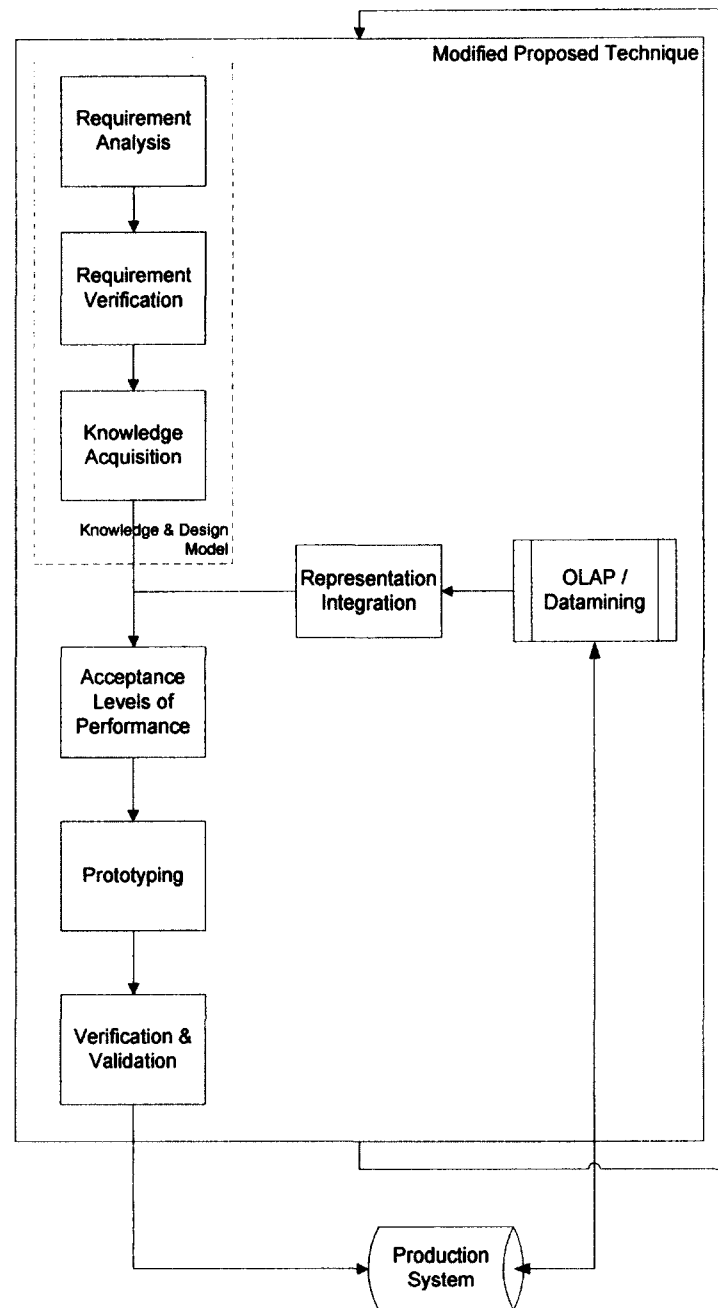


Figure 36. Modified technique for PONC Expert development

Results for Trillium scale are presented in Table 7. The overall score for this technique using this scale is 2.17.

Table 7.

Trillium scale results for proposed technique.

Stage	Observed characteristics
Problem specification	Level 2. Using knowledge model from commonKADS assures formal method for requirements, test and acceptance
Conceptual model	Level 3. A formal description is presented in knowledge model
Design model	Level 2. Design model only addresses knowledge part
Implemented model	Level 2. Spiral approach of iterations
Verification analysis	Level 2. Iteration approach based on spiral model ensures formal methods
Validation analysis	Level 2. Resulting system is evaluated on regular basis, either when an iteration is implemented or when new knowledge is included from Data Mining

In order to respond to questions 2 and 3, during the proposed development process new meta rules were identified as candidates and presented in a RO-RA-RV form, and verified against Table 4 criteria for contradictions and redundancy. These meta rules used to select candidate rules are illustrated in Table 8. The representation of RO-RA-RV for antecedent and consequent of this set of meta rules is presented in Table 9.

Table 8.

Meta rules used to select and map rules for PONC Expert, expressed in IF (LHS) THEN (RHS) form.

Rule	Antecedent (LHS)	Consequent (RHS)
D1	The field name of the rule is "ORDER NUMBER" and the field type of the rule is Numeric	The decision of the rule is accept AND the object name of the rule is order_detail AND the field name is order_number
D2	The field name of the rule is "ORDER NUMBER" AND (the field type of the rule is Bool OR field type of the rule is String)	The decision of the rule is reject
D3	The field name of the rule is "MATERIAL NAME" AND the field type is String	The decision of the rule is accept AND the object name of the rule is order_detail AND the field name is material
D4	The field name of the rule is "FACILITY NAME" AND the field type of the rule is String	The decision of the rule is accept AND the object name of the rule is eol AND the field name is facility
D5	The field name of the rule is "INTERIOR MATERIAL NAME" AND the field type of the rule is String	The decision of the rule is accept AND the object name of the rule is eol AND the field name is interior_material
D6	The field name of the rule is "THICKNESS" AND the field type of the rule is Numeric	The decision of the rule is accept AND the object name of the rule is eol AND the field name is thickness
D7	The field name of the rule is C DIVISION NAME AND the field type of the rule is String	The decision of the rule is accept AND the object name of the rule is order_detail AND the field name is division
D8	The field name of the rule is C REASON AND the field type of the rule is String	The decision of the rule is accept AND the object name of the rule is eol AND the field name is reason
D9	The field name of the rule is MATERIAL TYPE NAME AND the field type of the rule is String	The decision of the rule is accept AND the object name of the rule is eol AND the field name is material_type_name
D10	The field name of the rule is C LOCATION NAME AND the field type of the rule is String	The decision of the rule is accept AND the object name of the rule is eol AND the field name is location_name
D11	The field name of the rule is C ORDER TYPE DESCRIPTION AND the field type of the rule is String	The decision of the rule is accept AND the object name of the rule is order_detail AND the field name is orderType
D12	The field name of the rule is N PRODUCTION WEEK AND the field type of the rule is Numeric	The decision of the rule is accept AND the object name of the rule is order_detail AND the field name is production_week
D13	The field name of the rule is N COMPONENT QUANTITY AND the field type of the rule is Numeric	The decision of the rule is accept AND the object name of the rule is eol AND the field name is quantity
D14	The field name of the rule is PRODUCTION GROUP AND the field type of the rule is String	The decision of the rule is accept AND the object name of the rule is eol AND the field name is prod_group

Table 9.

Rule antecedent and consequent in standard RO-RA-RV form for meta rules.

	Rule antecedent in RO-RA-RV form	Rule consequent in RO-RA-RV form
D1	[=Field_name =Rule =ORDER NUMBER =Field_type =Rule =Numeric]	[=Decision =Rule =Accept =object_name =Rule =order_detail =field_name =Rule =order_number]
D2	[=Field_name =Rule =ORDER NUMBER =Field_type =Rule =Bool] [=Field_name =Rule =ORDER NUMBER =Field_type =Rule =Numeric]	[=Decision =Rule =Reject]
D3	[=Field_name =Rule =MATERIAL NAME =Field_type =Rule =String]	[=Decision =Rule =Accept =object_name =Rule =order_detail =field_name =Rule =material]
D4	[=Field_name =Rule =FACILITY NAME =Field_type =Rule =String]	[=Decision =Rule =Accept =object_name =Rule =eol =field_name =Rule =facility]
D5	[=Field_name =Rule =INTERIOR MATERIAL NAME =Field_type =Rule =String]	[=Decision =Rule =Accept =object_name =Rule =eol =field_name =Rule =interior_material]
D6	[=Field_name =Rule =THICKNESS =Field_type =Rule =Numeric]	[=Decision =Rule =Accept =object_name =Rule =eol =field_name =Rule =thickness]
D7	[=Field_name =Rule =C DIVISION NAME =Field_type =Rule =String]	[=Decision =Rule =Accept =object_name =Rule =order_detail =field_name =Rule =division]
D8	[=Field_name =Rule =C REASON =Field_type =Rule =String]	[=Decision =Rule =Accept =object_name =Rule =eol =field_name =Rule =reason]
D9	[=Field_name =Rule =MATERIAL TYPE NAME =Field_type =Rule =String]	[=Decision =Rule =Accept =object_name =Rule =eol =field_name =Rule =material_type_name]
D10	[=Field_name =Rule =C LOCATION NAME =Field_type =Rule =String]	[=Decision =Rule =Accept =object_name =Rule =eol =field_name =Rule =location_name]
D11	[=Field_name =Rule =C ORDER TYPE DESCRIPTION =Field_type =Rule =String]	[=Decision =Rule =Accept =object_name =Rule =order_detail =field_name =Rule =orderType]
D12	[=Field_name =Rule =N PRODUCTION WEEK DESCRIPTION =Field_type =Rule =Numeric]	[=Decision =Rule =Accept =object_name =Rule =order_detail =field_name =Rule =production_week]
D13	[=Field_name =Rule =N COMPONENT QUANTITY =Field_type =Rule =Numeric]	[=Decision =Rule =Accept =object_name =Rule =eol =field_name =Rule =quantity]
D14	[=Field_name =Rule =PRODUCTION GROUP =Field_type =Rule =String]	[=Decision =Rule =Accept =object_name =Rule =eol =field_name =Rule =prod_group]

With results in RO-RA-RV form and applying Table 4, no conflicting or redundant meta rules are detected. These rules are the initial iteration for the model described in Figure 37. Thus, they provide an implementation of a meta rule based system that integrates new rules and effectively translate them into production rules from an association model of a Data Mining repository for PONC Data Warehouse.

To address research questions 5 and 6, a description of the process of incorporating new rules obtained through Data Mining into PONC Expert is presented in Figure 38. A module for creating a Data Warehouse that host the Data Mining model and structure, and updating information from production database is required to initiate the process. Once established in a database server (MS SQL server 2005), an additional module was developed to present the resulting module in a graphical form to the user and allow rule extraction into an intermediate representation. A comma separated value file (CSV) with rules representation is used. This information is used by MetaRules module to generate rules and select those that are syntactically complete, creating another CSV file with rules explicitly written and input for an explanation. Finally, a separate module validates and incorporates these rules into PONC Expert knowledge base, allow the end user to incorporate this new knowledge in their decisions.

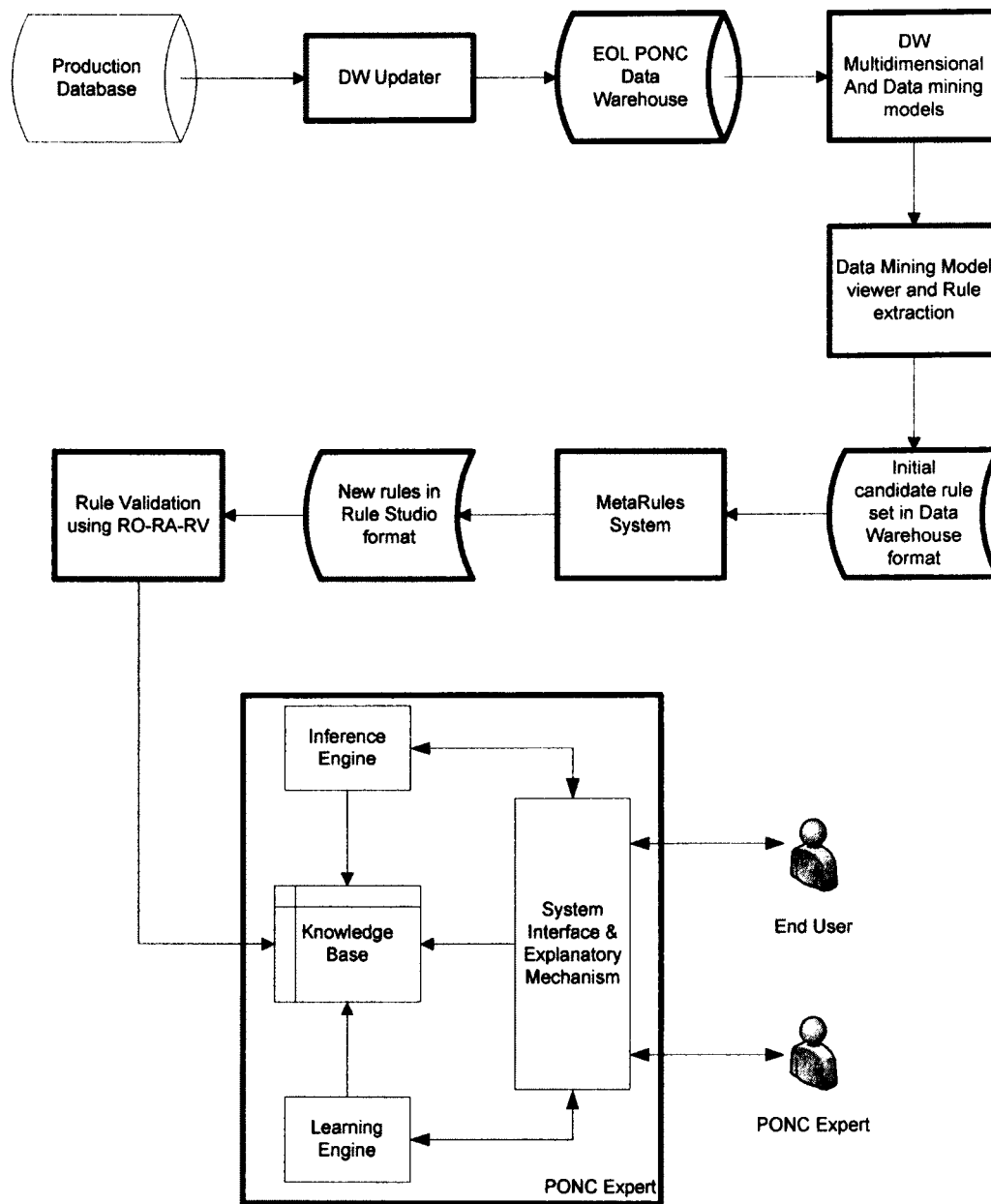


Figure 37. Data Mining new rules and its inclusion in PONC Expert

Results of evaluating PONC Expert over a randomly selected set of 50 PONC cases are presented in Table 10. Responses are presented following Table 5 format, as suggested by Mosquiera-Rey and Moret-Bonillo (2000). One source of answers (Source 1) correspond to PONC Expert as initially implemented, without incorporating new rules from Data Mining on it. A second source of responses (Source 2) is obtained with new rules added to PONC Expert, and an evaluation of results follows the analysis of agreement measurements as suggested by the authors using the agreement index. In this study, the comparison of two sources corresponds to the application of the ordinal scale of symbolic classification of a given interpretation between two intelligent systems and an expert response. Each interpretation of results ranges from very high (VH) agreement to very low agreement (VL) compared to the human expert answer.

Table 10.

PONC Expert pair comparison results.

		Source 2						
		VH	H	SH	N	SL	L	VL
Source 1	VH	5						
	H		1					
	SH							
	N							
	SL					1		
	L							
	VL	4		24			12	3

Based on these results the agreement index, the proportion of agreement and kappa index are calculated as described in equation 1 and 2:

$$p_0 = \frac{\sum_{i=j}^k n_{ij}}{N} = \frac{10}{50} = 0.2 \quad (4)$$

$$p_c = \sum_{i=j}^k p_i \cdot p_j = 0.0704 \quad (5)$$

$$Kappa = \frac{p_0 - p_c}{1 - p_c} = 0.1394$$

Kappa index significance is estimated as noted in equation 3, as follows:

$$Z_o = \frac{Kappa}{\sqrt{N(1-p_c)}} = 0.0204 \quad (6)$$

This value is contrasted with $Z_t = 1.96$ for a level of significance of 95%. Errors in this evaluation are considered of type I (builder risk) where a valid system is considered invalid and type II where an invalid system is considered valid as suggested by Mosqueira-Rey and Moret-Bonillo (2000).

Conclusions

Based on the initial research questions, the following conclusions are obtained from collected data:

1. A formal technique is proposed by combining commonKADS principles of knowledge model and design model into a spiral development cycle, as illustrated in Figure 36. The final score obtained from Table 7 for this proposed technique is 2.17, which is significantly higher than 2 in the Trillium scale for a formal analysis as initially stated. Therefore, the technique can be used as a formal method to develop knowledge based solutions.
2. The proposed technique used in the development of PONC Expert and compared with a traditional commonKADS technique addresses structural errors which are presented in the analysis of meta rules using RO-RA-RV form rules to detect redundancy and contradictions. Furthermore, every rule obtained from Data Mining and incorporated on the implementation instance of PONC Expert is already in the AND-OR form noted in Table 8 and 9. Simple inspections of the resulting rule set in the RO-RA-RV form allow detecting redundancy or contradictory new rules to incorporate using their antecedent and consequent form.
3. Meta rules were successfully used to integrate new rules obtained from an association model over PONC department assigned in PONC Expert. Although used to determine syntax correctness for resulting rules and translation of rules from Data Mining form and interpretation into domain problem notation, the use of meta rules also helps to

determine the position of the new rule in the resulting new knowledge based representation set.

4. The system architecture presented in Figure 37 represents a working model of the domain problem application of the proposed technique and its interaction among different implemented modules, specifically the mechanism to incorporate new rules derived from Data Mining models into the knowledge based initial solution. In the evaluation of response to 50 cases randomly selected from different departments, the agreement index was significantly low and suggests that the new rules incorporated into the Expert System increase its potential use by providing more answers to different cases if compared to the initial solution based on traditional methods.
5. In the design of the experiment, an Expert System was developed using traditional methods. Using Data Mining models and meta rules, new knowledge was incorporated and responses from both systems were compared using an agreement and a Kappa index. An initial iteration produced 68 new rules to be incorporated into the system. Responses were significantly different between both solutions, as presented in Equations 4 and 5. When calculating the significance of the Kappa index in Equation 6, it resulted to be 0.0204 which is significantly lower than the expected value of 1.96. This result indicates that the new system that incorporated the additional rules obtained from Data Mining provide responses that are different than the ones obtained from the initial system.

Data Mining resulting rules notably changed the system from its original form, providing more responses on cases where the initial system was unable to conclude. Furthermore, the system was able to respond in 28 more cases than the original system.

Recommendations

The following recommendations are made as a result of this study:

The application of a modified technique in the automation of knowledge integration obtained from Data Warehouses into an existing knowledge based program is a promising field for further research. It allows organizations to identify and incorporate valuable knowledge existing in current transactional systems and enhance knowledge based system responses. Although a simple prototype was developed, it can be extended to incorporate several aspects where opportunities for enhancements appear.

Replication of the current research with a different sample size is suggested to verify the process and results obtained in this study. The replication in other problem domains is also recommended.

An area for further research is the development of methodologies that detect and incorporate meta knowledge in knowledge automation. Current research focus on production rules but little research has been done in meta knowledge since Mycin and Meta Dendral projects were developed. Most of the area of knowledge elicitation is oriented to first order logic only. An adaptation of existing techniques was introduced as an initial step towards a more robust body of knowledge in this area.

An opportunity for improvement is the inclusion and automated verification of selected rules into RO-RA-RV form. Although it covered meta rules for consistency and redundancy, it can also be extended to automate this step for new production rules obtained from Data Mining models.

An aspect of particular interest for future studies is the analysis of different Data Mining models and its impact in the final solution. Only one model was implemented, association rules on a simplified structure. The use of more sophisticated Data Mining models or complex structures may have an impact in the response of the modified knowledge system.

The system architecture presented in Figure 37 can be extended to fully automate the new knowledge incorporation process. Although it comprised several modules, they can be integrated into multiple agents and coordinate their state in an intelligent structure using meta rules.

Fuzzy logic techniques can be used instead of meta rules to derive new knowledge representation instances and incorporate them into an existing Expert System.

REFERENCES

- Abbass, H. A., Towsey, M., Finn, G., & Kozan, E. (2001). A meta-representation for integrating operational research and artificial intelligence in an intelligent decision support paradigm. *International Journal in Operational Research*, 8, 107-119.
- Akkermans, H., Van de Velde, W., Wielinga, B., & Schreiber, G. (1993). *Expertise model definition document* (Deliverable ESPRIT Nr. P5248 KADS-II/M2/UvA/0026/1.1. Rapport aan: EC.). Retrieved March 2004 from <http://swi.psy.uva.nl/projects/CommonKADS/postscript/CK-UvA-26.ps.gz>
- Alavis, M. (1999). Knowledge management systems: Issues, challenges and benefits. *Communications of the Association for Information Systems*, 1(7), 1-37.
- Albion Research Ltd. (2011). Retrieved November, 2011, from http://www.albionresearch.com/data_mining/market_basket.php
- Anthes, G. (2009). AI comes of age. *Computerworld*, 43, 16-19.
- Back, A., Von Krogh, G., Seufert, A., & Enkel, E. (2005). *Putting knowledge networks into action*. Berlin, Germany: Springer-Verlag.
- Beckman, T. J. (1999). The current state of knowledge management. In J. Liebowitz (Ed.), *Knowledge management handbook* (pp. 1.1-1.22). Boca Raton, CA: CRC Press.
- Benfer, R. A., Furbee, L., & Brent, E. E. (1996). Expert systems and the representation of knowledge. *American Anthropological Association*, 23(2), 416-420.
- Brackenbury, I., & Ravin, Y. (2002, September). Machine intelligence and the Turing Test. (Technical forum) [Electronic version]. *IBM Systems Journal*, 41(3), 524-529.
- Bronzino, J. D., & Morelli, R. A. (1989). *Expert System: Basic concepts*. New York: NLA Monograph Series.
- Chandrasekaran, B., Johnson, T. R., & Smith, J. W. (1992). Task-structure analysis for knowledge modeling. *Communications of the ACM*, 35(9), 124-137.
- Christiansen, H., & Martinenghi, D. (2000). Symbolic constraints for meta-logic programming. *Applied Artificial Intelligence*, 14, 345-367.
- Cody, W. F., Kreulen, J. T., Krishna, V., & Spangler, W. S. (2002). The integration of business intelligence and knowledge management. *IBM Systems Journal*, 41(4), 697-713.

- Craig, I. D. (1991). *Formal specification of advanced AI architectures*. New York: Ellis Horwood.
- Cranefield, S., & Purvis, M. (1999). *UML as an Ontology modelling language*. Paper presented at the meeting of the Proceedings of the IJCAI-99, Workshop of Intelligent Information Integration. Dunedin, New Zealand.
- Crosby, P. (Writer and Director). (2004). *Making quality clear* [videocassette]. (Philip Crosby Associates, 306 Dartmouth Street, Boston, MA 02116).
- Dieng, R., Croby, O., & Lapalut, S. (1995). Acquisition and exploitation of gradual knowledge. *International Journal of Human-Computer Studies*, 42, 465-499.
- Dieste, O., Juristo, N., Moreno, A. M., Pazos, R., & Sierra, A. (2002). Conceptual modeling in software engineering and knowledge engineering: Concepts, techniques and trends. In S. K. Chang (Ed.), *Handbook of Software Engineering and Knowledge Engineering* (pp. 733-766). River Edge, NJ: World Scientific Publishing Company.
- Espil, M. M., & Vaisman, A. A. (2003). Revising aggregation hierarchies in OLAP: A rule-based approach. *Data & Knowledge Engineering*, 45, 225-256. Retrieved May 1, 2006, from Elsevier Science B.V. Web Site: <http://www.ComputerScienceWeb.com>
- Fensel, D., Groenboom, R., & Renardel de Lavalette, G. R. (1998). Modal Change Logic (MCL): Specifying the reasoning of knowledge-based systems. *Data & Knowledge Engineering*, 26, 243-269.
- Gennari, J. H., Musen, M. A., Ferguson, R. W., Grosso, W. E., Crubezy, M., Eriksson, H., et al. (2003). The evolution of Protege: an environment for knowledge-based system development. *International Journal of Human-Computer Studies*, 58, 89-123.
- Giarratano, J., & Riley, G. (1998). *Expert Systems. Principles and programming* (3rd ed.). Boston, MA: PWS Publishing Company.
- Gorla, N. (2003). Features to consider in a data warehousing system. *Communications of the ACM*, 46(11), 111-115.
- Guckenheimer, S. & Perez, J. J. (2006). *Software Engineering with Microsoft Visual Studio Team System*. Available from http://books.google.com/books?id=D8b7RDqde_EC&source=gbs_atb
- Hamfelt, A., & Hansson, A. (1991. April 20). *Metalogic representation of stratified knowledge* (UPMAIL Technical Report No. 66). Uppsala, Sweden: Uppsala University.

- Haouche-Gingins, C., & Charlet, J. (1998). A knowledge-level testing method. *International Journal of Human-Computer Studies*, 49, 1-20.
- Harmon, P., & Sawyer, B. (1990). *Creating Expert Systems for business and industry*. New York: John Wiley & Sons, Inc.
- Heliades, G. P., & Edmonds, E. A. (1999). On facilitating knowledge transfer in software design. *Knowledge-Based Systems*, 12, 391-395.
- Herrmann, T., Kienle, A., & Reiband, N. (2003). Meta-knowledge. A success factor for computer-supported organizational learning in companies. *Educational Technology & Society*, 6(1), 9-13.
- Hicks, R. C. (2003). Knowledge base management systems-tools for creating verified intelligent systems. *Knowledge-Based Systems*, 16, 165-171.
- Hoffmann, A. (1998). *Paradigms of Artificial Intelligence*. Singapore: Springer-Verlag.
- Hopgood, A. A. (2003, May). Artificial intelligence: Hype or reality? *Computer*, 36, 24-28.
- Huang, C. J., & Cheng, M. Y. (2008). Conflicting treatment model for certainty rule-based knowledge. *Expert Systems with Applications*, 35, 161-176. Retrieved April 1, 2011, from Elsevier Ltd Web Site: <http://www.sciencedirect.com>
- Jackson, P. (1985). *Introduction to Artificial Intelligence*. New York: Dover Publications.
- Jackson, P. (1999). *Introduction to Expert Systems* (3rd ed.). Harlow, England: Addison-Wesley.
- Jensen, M. R., Moller, T. H., & Pedersen, T. B. (2001). Specifying OLAP cubes on XML data. *Journal of Intelligent Information Systems*, 17(2-3), 255-280.
- Juristo, N., & Morant, J. L. (1998). Common framework for the evaluation process of KBS and conventional software. *Knowledge-Based Systems*, 11, 145-159.
- Kalfoglou, Y., Menzies, T., Althoff, K. D., & Motta, E. (2000). Meta-knowledge in system design: panacea or undeliverable promise? *The Knowledge Engineering Review*, 15(4), 381-404.
- Kantardzic, M. (2003). *Data Mining. Concepts, model, methods and algorithms*. Piscataway, NJ: IEEE Press.
- Kerlinger, F. N. (1992). *Foundations of behavioral research* (3rd ed.). Fort Worth, FL: Harcourt Brace College Publishers.

- Kimball, R., & Merz, R. (2000). *The Data Warehouse toolkit*. New York: Wiley Computer Publishing.
- King, W. R., Marks, P. V., & McCoy, S. (2002). The most important issues in knowledge management. *Communications of the ACM*, 45(9), 93-97.
- Larose, D. (2006). *Data Mining. Methods and models*. Hoboken, NJ: John Wiley & Sons, Inc.
- Lavene, M., & Loizou, G. (2003). Why is the snowflake schema a good Data Warehouse design? *Information Systems*, 28, 225–240.
- Lechtenborger, J., & Vossen, G. (2003). Multidimensional normal forms for Data Warehouse design. *Information Systems*, 28, 415-434.
- Lee, C. Y. (2003). A knowledge management scheme for meta-data: An information structure graph. *Decision Support Systems*, 1055, 1-14. Retrieved March 10, 2005, from Elsevier Science B.V. Web Site: <http://www.sciencedirect.com>
- Le Goc, M., Frydman, C., & Torres, L. (2002). Verification and validation of the SACHEM conceptual model. *International Journal of Human-Computer Studies*, 56, 199-223. Retrieved October 2, 2003, from <http://www.idealibrary.com>
- Lenat, D. (2008). The voice of the turtle: whatever happened to AI? [Electronic version]. *AI Magazine*, 29(2), 11-22.
- Ma, C., Chou, D. C., & Yen, D. C. (2000). Data Warehousing, technology assessment and management. *Industrial Management & Data System*, 100(3), 125-134.
- Martinez-Bermudez, I. (2001). A framework for knowledge acquisition, representation and problem-solving in knowledge-based planning (Doctoral dissertation, Michigan State University). *Dissertation Abstracts International*, 62, 03.
- Matison, R. (1999). *Web warehousing and knowledge management*. New York: McGraw-Hill.
- Meseguer, P., & Plaza, E. (1992). The Valid project: Goals, development, and results. *International Journal of Intelligent Systems*, 9 (9), 867-892.
- Moeller, R. A. (2001). *Distributed data warehousing using web technology*. New York: Amacom.
- Mosqueira-Rey, E., & Moret-Bonillo, V. (2000). Validation of intelligent systems: A critical study and a tool. *Expert Systems with Applications*, 18, 1-16.

- Nabil, A. (1999, February). *Development of an Expert System*. Retrieved February 1, 1999, from The State University of New Jersey Web Site: <http://civic3.rutgers.edu/~adam/courses/es/node8.html>
- Negnevitsky, M. (2005). *Artificial Intelligence. A guide to Intelligent Systems* (2nd ed.). Harlow, England: Addison-Wesley.
- Nemati, H. R., Steiger, D. M., Iyer, L. S., & Herschel, R. T. (2002). Knowledge warehouse: An architectural integration of Knowledge Management, Decision Support, Artificial Intelligence and Data Warehouse. *Decision Support Systems*, 33, 143-161.
- Nettleton, D., & Muñiz, J. (2001). Processing and representation of meta-data for sleep apnea diagnosis with an artificial intelligence approach. *International Journal of Medical Informatics*, 63, 77-89.
- Noy, N. F., & McGuinness, D. L. (2001). *Ontology development 101: A guide to creating your first ontology* (Stanford Knowledge Systems Laboratory KSL-01-05). Retrieved March 2004 from <http://www.ksl.stanford.edu/people/dlm/papers/ontology-tutorial-noy-mcguinness.pdf>
- Openshaw, S., & Openshaw, C. (1997). *Artificial Intelligence in Geography*. Chichester, England: John Wiley & Sons Ltd.
- Oussalah, M. (2003). Reuse in KBS: A component approach. *Expert Systems with Applications*, 24, 173-181.
- Paradigm. A definition (2003). *Encarta encyclopedia* [CD-ROM]. Redmont, WA: Microsoft Corporation.
- Pedersen, T. B., Jensen, C. S., & Dyreson, C. E. (2001). A foundation for capturing and querying complex multidimensional data. *Information Systems*, 26, 383-423.
- Pendse, N. (2001). The ins and outs of olap. *Computer Weekly*, 26.
- Plant, R. T., & Gamble, R. (1997). Using meta-knowledge within a multilevel framework for KBS development. *International Journal of Human-Computer Studies*, 46, 523-547.
- Raisanen, R., & Oinas-Kukkonen, H. (2008). A system architecture for the 7C knowledge environment. *Information Modeling and Knowledge Bases*, XIX, 217-236. Retrieved February 25, 2011, from University of Northern Iowa Web Site: <http://site.ebrary.com/lib/rodlibrary/Doc?id=10216840&ppg=248>.

- Rech, J., Ras, E., & Decker, B. (2007). Riki: A system for knowledge transfer and reuse in software engineering projects. In M. Lytras & A. Naeve (Eds.), *Open Source for knowledge and learning management: Strategies beyond tools* (pp. 52-121). Hershey, PA: Idea Group Publishing.
- Reich, Y. (1996). Artificial intelligence in bridge engineering. *Microcomputers in Civil Engineering*, 10(2), 433-445.
- Russel, S. J., & Norvig, P. (2003). *Artificial intelligence. A modern approach*. Upper Saddle River, NJ: Prentice Hall.
- Schreiber, G. (2010). *CommonKADS, Course slides*. Retrieved July 10, 2010, from CommonKADS, Engineering and Managing Knowledge Web Site: <http://www.commonkads.uva.nl/frameset-commonkads.html>
- Shim, J. P., Warkentin, M., Courtney, J. F., Power, D. J., Sharda, R., & Carlsson, C. (2002). Past, present, and future of decision support technology. *Decision Support Systems*, 33, 111-126.
- Speel, P. H., & Aben, M. (1998). Preserving conceptual structures in design and implementation of industrial KBSs. *International Journal of Human-Computer Studies*, 49, 547-575.
- Spiegler, I. (2000). Knowledge management: A new idea or a recycled concept? *Communications of the association for Information Systems*, 3(14), 1-24.
- SQL Server 2005 Books Online (2008). Data Mining algorithms. Retrieved November, 2011, from [http://technet.microsoft.com/en-us/library/ms175595\(SQL.90\).aspx](http://technet.microsoft.com/en-us/library/ms175595(SQL.90).aspx)
- Thalhammer, T., Schrefl, M., & Mohania, M. (2001). Active Data Warehouse: Complementing OLAP with analysis rules. *Data & Knowledge Engineering*, 39, 241-269.
- Thomsen, E. (2002). *OLAP solutions: Building multidimensional information systems* (2nd ed.). Chichester, NY: Wiley.
- Tommelein, I. D., Levitt R. E., & Hayes-Roth, B. (1992). Site-Layout Modeling: How can Artificial Intelligence help? *Journal of Construction Engineering and Management*, 118, 594-611.
- Umstatt, W. D., & Davis, C. W. (2000). *Modern cabinetmaking*. Tinley Park, Illinois: The Goodheart-Willcox Company, Inc.
- Valente, A., Breuker, J., & Van de Velde, W. (1998). The CommonKADS library in perspective. *International Journal of Human-Computer Studies*, 49, 391-416.
- Walter, J. (2003, April). The ins and outs of Data Mining. *R&D*, 45, 33-34.

- Wang, H., & Wang, S. (2008). A knowledge management approach to Data Mining process for business intelligence. *Industrial Management & Data Systems*, 108(5), 622-634. Retrieved June 1, 2010, from <http://www.emeraldinsight.com/0263-5577.htm>
- Weitzel, J. R., & Kerschberg, L. (1989). Developing knowledge based systems: Reorganizing the system development life cycle. *Communications of the ACM*, 32(4), 482-488.
- Wentworth, J. A., Knaus, R., & Aougab, H. (1995). Validating underlying knowledge. In *Verification, validation and evaluation of Expert Systems, volume I, A FHWA Handbook* (chap. 9). Retrieved March 3, 2001, from Federal Highway Administration. U.S. Department of Transportation Web Site: <http://www.fhwa.dot.gov/publications/research/safety/aard/index.cfm#val>
- Wielinga, B.J., Akkermans, J.M., Schreiber, A.Th. (1993). Validation and Verification of Knowledge Models. In J. Cardenosa, P. Meseguer, (eds.); *EUROVAV'93: Proceedings of the European Symposium on the Validation and Verification of Knowledge-Based Systems*, Universidad Politecnica de Madrid, Palma de Mallorca, p. 29-50.
- Wiig, K. M. (1999). Introducing knowledge management into the enterprise. In J. Liebowitz (Ed.), *Knowledge management handbook* (pp. 3.1-3.41). Boca Raton, CA: CRC Press.

APPENDIX A
PONC EXPERT SOURCE CODE

PONCExpert/frmMain.frm

```

Attribute VB_Name = "frmMain"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
Option Explicit

Dim mrs As New RSPONC
Dim mbFirstLoad As Boolean
Dim miCount As Integer
Dim mQryObj As New QueryObject

Private Sub cmdExit_Click()
    mQryObj.m_qryPending = False
    Dim f As Form
    For Each f In Forms
        Unload f
    Next
End Sub

Private Sub cmdProcess_Click()
    mQryObj.m_qryPending = False
    lblConclusion.Text = lblConclusion.Text & vbNewLine & txtQuestion
    lblConclusion.Text = lblConclusion.Text & vbNewLine & "=> "
    If (queryType = vrms.rmString Or queryType = vrms.rmNumeric Or
        queryType = vrms.rmDateTime) Then
        lblConclusion.Text = lblConclusion.Text & ValueText.Text
    ElseIf queryType = vrms.rmSimple Then
        lblConclusion.Text = lblConclusion.Text + CStr(OptTrue.Value)
    ElseIf queryType = vrms.rmCompound Then
        lblConclusion.Text = lblConclusion.Text + ValueList.Text
    End If
End Sub

Private Sub cmdReset_Click()
    miCount = miCount + 1
    mrs.Reset
    mrs.Engine.Reset

    *** Set query object to our local object
    mrs.QueryObject = mQryObj

    *** Clean up the GUI
    lblConclusion.Text = ""
    Label1.Enabled = True
    Label2.Enabled = True
    txtQuestion.Enabled = True
    cmdProcess.Enabled = True
    lblConclusion.Enabled = True

    Dim x As String
    x = mrs.Classes.EOL.suggested_action
    If Len(x) > 0 Then

```

```

        lblConclusion.Text = lblConclusion.Text & vbCrLf & "Explanation:" &
vbNewLine & mrs.Classes.EOL.explanation & vbCrLf & vbCrLf &
"Instructions:" & vbCrLf & mrs.Classes.EOL.suggested_action
    Else
        lblConclusion.Text = lblConclusion.Text & vbCrLf & "Unable to
conclude based on response" & vbCrLf & "Try again with another PONC"
& vbCrLf
    End If
    'miCount = miCount - 1

    'If miCount = 0 Then
    '    '** Clean up the GUI
    '    Label1.Enabled = False
    '    Label2.Enabled = False
    '    txtQuestion.Enabled = False
    '    'cmdProcess.Enabled = False
    '    lblConclusion.Enabled = True
    '    'lstResults.Text = lstResults.Text & vbCrLf & "*** " &
vbNewLine & lblConclusion
    'End If
End Sub

Private Sub Form_Activate()

    If mbFirstLoad = True Then
        mbFirstLoad = False
        cmdReset_Click
    End If

End Sub

Private Sub Form_Load()
    Dim numReason As Integer
    Dim i As Integer
    Dim buff As String
    Dim linesplit() As String

    miCount = 0
    mbFirstLoad = True
    '** Reset the rules set...
    mrs.Reset
    '** Set query object to our local object
    mrs.QueryObject = mQryObj
    Set mQryObj.m_rs = mrs

    '** load all reason codes for PONC from file
    Open App.Path & "\PONC_REASON.CSV" For Input As #1
    Line Input #1, buff
    buff = ""
    i = 0
    While Not EOF(1) ' check if at end of file
        Line Input #1, buff
        ReDim Preserve rsReason(i)
        rsReason(i) = buff
        i = i + 1
    Wend

```



```
Close #1

Set prompt = New Dictionary
Open App.Path & "\PONC_QUESTION.CSV" For Input As #1
Line Input #1, buff
buff = ""
While Not EOF(1)
    Line Input #1, buff
    linesplit = Split(buff, ",")
    prompt.Add linesplit(0), linesplit(1)
Wend
Close #1

End Sub
```

PONCExpert/QueryObject.cls

```

Attribute VB_Name = "QueryObject"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = True
Attribute VB_PredeclaredId = False
Attribute VB_Exposed = False
Option Explicit
Implements vrms.Queries

Private m_ObjectName As String
Private m_AttributeName As String
Private m_AttributeClass As vrms.Attribute
Private m_AttributeObject As Object
Private Const QyeryCaption = "Default QueryObject - Enter value for:"
Public m_qryPending As Boolean
Public m_rs As Object

Private Sub controlShow(controlName As String)
    frmMain.ValueList.Visible = (controlName = "Compound")
    frmMain.ValueList.Clear

    frmMain.frameSimple.Visible = (controlName = "frameSimple")
    frmMain.OptTrue.Value = False
    frmMain.OptFalse.Value = False

    frmMain.ValueText.Visible = (controlName = "String" Or controlName =
"Numeric" Or controlName = "DateRange")
    frmMain.ValueText.Text = ""
End Sub

Private Sub doCompoundQuery()
    Dim itemsList As String
    Dim idx As Long
    Dim compItems As vrms.CompoundItems
    Dim compItem As vrms.CompoundItem

    Dim pos As Variant
    Dim start As Long
    Dim tStr As String

    Set compItems = m_AttributeClass.CompoundItems
    For idx = 0 To compItems.Count - 1
        Set compItem = compItems.Item(idx)
        itemsList = itemsList + compItem.Name + ","
    Next

    controlShow "Compound"
    queryType = vrms.rmCompound

    start = 1
    pos = InStr(start, itemsList, ",")
    While (pos <> 0)
        tStr = Mid(itemsList, start, pos - start)
        frmMain.ValueList.AddItem tStr
        start = pos + 1
    End While

```

```

    pos = InStr(start, itemsList, ",")
Wend

If prompt.Exists(m_ObjectName + "." + m_AttributeName) Then
    frmMain.txtQuestion.Text = prompt.Item(m_ObjectName + "." +
m_AttributeName)
Else
    frmMain.txtQuestion.Text = m_ObjectName + "." + m_AttributeName
End If

m_gryPending = True
While (m_gryPending = True)
    DoEvents
    Sleep (100)
Wend

*** Not a good solution, temporary....
If Not (m_AttributeObject Is Nothing) Then
    If (queryType = vrms.rmCompound) Then
        m_AttributeObject = frmMain.ValueList.Text
    End If
End If

End Sub

Private Sub doDateTimeQuery()
    'm_AttributeObject = StringQueryDlg(m_ObjectName + "." +
m_AttributeName, QyeryCaption)
    If prompt.Exists(m_ObjectName + "." + m_AttributeName) Then
        frmMain.txtQuestion.Text = prompt.Item(m_ObjectName + "." +
m_AttributeName)
    Else
        frmMain.txtQuestion.Text = m_ObjectName + "." + m_AttributeName
    End If

    controlShow "DateRange"
    queryType = vrms.rmDateTime

    m_gryPending = True
    While (m_gryPending = True)
        DoEvents
        Sleep (100)
    Wend

    *** Not a good solution, temporary....
    If Not (m_AttributeObject Is Nothing) Then
        If queryType = vrms.rmDateTime Then
            m_AttributeObject = frmMain.ValueText.Text
        End If
    End If

End Sub

Private Sub doNumericQuery()
    If prompt.Exists(m_ObjectName + "." + m_AttributeName) Then
        frmMain.txtQuestion.Text = prompt.Item(m_ObjectName + "." +

```

```

m_AttributeName)
Else
    frmMain.txtQuestion.Text = m_ObjectName + "." + m_AttributeName
End If

controlShow "Numeric"
queryType = vrms.rmNumeric

m_qryPending = True
While (m_qryPending = True)
    DoEvents
    Sleep (100)
Wend

''' Not a good solution, temporary....
If Not (m_AttributeObject Is Nothing) Then
    If queryType = vrms.rmNumeric Then
        m_AttributeObject = frmMain.ValueText.Text
    End If
End If
End Sub

Private Sub doSimpleQuery()

controlShow "frameSimple"
queryType = vrms.rmSimple

If prompt.Exists(m_ObjectName + "." + m_AttributeName) Then
    frmMain.txtQuestion.Text = prompt.Item(m_ObjectName + "." +
m_AttributeName)
Else
    frmMain.txtQuestion.Text = m_ObjectName + "." + m_AttributeName
End If

queryType = vrms.rmSimple

m_qryPending = True
While (m_qryPending = True)
    DoEvents
    Sleep (100)
Wend

''' Not a good solution, temporary....
If Not (m_AttributeObject Is Nothing) Then
    If (queryType = vrms.rmSimple) Then
        m_AttributeObject = CStr(frmMain.OptTrue.Value)
    End If
End If

End Sub
Private Sub doStringQuery()
    'm_AttributeObject = StringQueryDlg(m_ObjectName + "." +
m_AttributeName, QueryCaption)
    If prompt.Exists(m_ObjectName + "." + m_AttributeName) Then
        frmMain.txtQuestion.Text = prompt.Item(m_ObjectName + "." +
m_AttributeName)

```

```

Else
    frmMain.txtQuestion.Text = m_ObjectName + "." + m_AttributeName
End If

controlShow "String"
queryType = vrms.rmString

m_gryPending = True
While (m_gryPending = True)
    DoEvents
    Sleep (100)
Wend

*** Not a good solution, temporary....
If Not (m_AttributeObject Is Nothing) Then
    If (queryType = vrms.rmNumeric Or queryType = vrms.rmString) Then
        m_AttributeObject = frmMain.ValueText.Text
    End If
End If
End Sub

Public Sub Queries_Query(ByVal ObjectName As String, ByVal
AttributeClass As String, ByVal AttributeObject As Object)

    m_ObjectName = ObjectName
    m_AttributeName = AttributeName
    Set m_AttributeClass = AttributeClass
    Set m_AttributeObject = AttributeObject

    Select Case AttributeClass.DataType
        Case Is = vrms.rmSimple
            Call doSimpleQuery
        Case Is = vrms.rmNumeric
            Call doNumericQuery
        Case Is = vrms.rmString
            Call doStringQuery
        Case Is = vrms.rmDateTime
            Call doDateTimeQuery
        Case Is = vrms.rmCompound
            Call doCompoundQuery
    End Select
    Set m_AttributeClass = Nothing
    Set m_AttributeObject = Nothing
    Set AttributeClass = Nothing
    Set AttributeObject = Nothing

End Sub

```

PONCExpert/Win32Decls.bas

```
Attribute VB_Name = "Win32Decls"
```

```
***
```

```
*** Win32 API function decls
```

```
***
```

```
Declare Sub Sleep Lib "kernel32" (ByVal dwMilliseconds As Long)
```

```
Public queryType As String
```

```
Public rsReason() As String
```

```
Public prompt As Dictionary
```

```
***
```

```
*** Sub Main
```

```
***
```

```
Public Sub Main()
```

```
    Load frmMain
```

```
    frmMain.Show
```

```
End Sub
```

PONCExpert/RSPONC.Dsr

```

!=====
! CLASS Declarations
Class EOL
With suggested_action Compound keep,reassign_to_IS,
reassign_to_DDG,reassign_to_RND,
reassign_to_CDS,reassign_to_BDS,reassign_to_finish_mill,reassign_to_rou
gh_mill,
not_a_ponc, reassign_to_D21
Search Order Context Rules Default
With explanation String Default " "
Search Order Context Rules Default

With value Numeric
With assigned_to String Default " "
With order_number Numeric
With order_line Numeric
With comment String
With component_changed Simple
With width Numeric
With height Numeric
With thickness Numeric
With reason String
With facility String
With material String
With interior_material String
With material_type_name String
With location_name String
With quantity Numeric
With prod_group String

Class cut_list
With division String
With series String
With cutlist_mill String
With facility String
With component_po Numeric
With order_number Numeric
With report_name String
With date_printed Time

Class order_detail
With order_number Numeric
With order_line Numeric
With style String
With material String
With color String
With series String
With division String
With glaze String
With componentPO Numeric
With reprocess_date Time
With catalog String
With production_week Numeric
With orderType String
!!With orderType Compound

```

normal, replacement, job_site, literature, rga, display, fast_trax, ups, sample
, backorder, shipping_copies

Class PO

With componentPO Numeric

With date_release Time

With division String

Class notification

With notification_date Time

With title String

With notification_list String

With content String

Class assembly_sheet

With order_number Numeric

With order_line Numeric

With componentPO Numeric

With bundle Numeric

Class modification

With last_change_date Time

With content String

With has_logical_error Simple

Class formula

With last_change_date Time

With last_route_change Time

With component String

!=====

! RULE & DEMON Declarations

Rule 20

If UpCase(EOL.reason)="NOT ON ASSEMBLY SHEET"

And EOL.component_changed=True

And EOL.assigned_to <> "DDG"

Then EOL.suggested_action Is reassign_to_DDG

And EOL.explanation := "Component was changed but not included
initially on assembly sheets."

Rule 30

If UpCase(EOL.reason)="NOT ON ASSEMBLY SHEET"

And EOL.component_changed=True

And EOL.assigned_to = "DDG"

Then EOL.suggested_action Is keep

And EOL.explanation := "PONC is already assigned to the correct
department"

Rule 40

If UpCase(EOL.reason) = "NOT ON CUT LIST"

And EOL.component_changed = True

And EOL.assigned_to <> "CDS"

And UpCase(order_detail.division) = "CUSTOM"

And order_detail.reprocess_date > PO.date_release

Then EOL.suggested_action Is reassign_to_CDS

And EOL.explanation := "Changes were made to the order after it was

released for production"

Rule 50

```
If UpCase(EOL.reason) = "NOT ON CUT LIST"
  And EOL.component_changed = True
  And EOL.assigned_to = "CDS"
  And UpCase(order_detail.division) = "CUSTOM"
  And order_detail.reprocess_date > PO.date_release
Then EOL.suggested_action Is keep
  And EOL.explanation := "Component was changed and notified
accordingly"
```

Rule 60

```
If UpCase(EOL.reason) = "NOT ON CUT LIST"
  And EOL.component_changed = True
  And EOL.assigned_to <> "BDS"
  And UpCase(order_detail.division) = "BATH"
  And order_detail.reprocess_date > PO.date_release
Then EOL.suggested_action Is reassign_to_BDS
  And EOL.explanation := "Item was scheduled in BDS and then rescheduled
after paperwork was printed"
```

Rule 70

```
If UpCase(EOL.reason) = "NOT ON CUT LIST"
  And EOL.component_changed = True
  And EOL.assigned_to = "BDS"
  And UpCase(order_detail.division) = "BATH"
  And order_detail.reprocess_date > PO.date_release
Then EOL.suggested_action Is keep
  And EOL.explanation := "Already assigned to BDS. Changes were made
after it was scheduled"
```

Rule 80

```
If UpCase(EOL.reason)= "NOT ON CUT LIST"
  And EOL.component_changed=False
  And EOL.assigned_to <> "DDG"
  And notification.notification_date >= cut_list.date_printed
Then EOL.suggested_action Is reassign_to_DD
  And EOL.explanation := "Order may have been reprocessed and changes
were not reflected on paperwork"
```

Rule 90

```
If UpCase(EOL.reason)= "NOT ON CUT LIST"
  And EOL.component_changed=False
  And EOL.assigned_to = "DDG"
  And notification.notification_date >= cut_list.date_printed
Then EOL.suggested_action Is keep
  And EOL.explanation := "Already assigned to the correct department.
Changes were notified and no printlist was created"
```

Rule 100

```
If UpCase(EOL.reason) = "CUT LIST WRONG"
  And EOL.component_changed=False
  And EOL.assigned_to <> "RND"
  And modification.last_change_date > PO.date_release
Then EOL.suggested_action Is reassign_to_RND
```

And EOL.explanation := "Modifications on the order were changed after the component was scheduled on PO"

Rule 110

```
If UpCase(EOL.reason) = "CUT LIST WRONG"
  And EOL.component_changed=False
  And EOL.assigned_to = "RND"
  And modification.last_change_date > PO.date_release
Then EOL.suggested_action Is keep
  And EOL.explanation := "Item is already assigned to the correct
department"
```

Rule 120

```
If UpCase(EOL.reason) = "CUT LIST WRONG"
  And EOL.component_changed = True
  And EOL.assigned_to <> "RND"
  And modification.last_change_date < PO.date_release
Then EOL.suggested_action Is reassign_to_RND
  And EOL.explanation := "Mod needs to be changed"
```

Rule 130

```
If UpCase(EOL.reason) = "CUT LIST WRONG"
  And EOL.component_changed = True
  And EOL.assigned_to = "RND"
  And modification.last_change_date < PO.date_release
Then EOL.suggested_action Is keep
  And EOL.explanation := "Mod needs to be modified and PONC is assigned
to RND"
```

Rule 140

```
If (UpCase(EOL.reason) = "LIST ERROR" Or UpCase(EOL.reason) = "NOT ON
ASSEMBLY SHEET")
  And Searchstr("SHIP-LOOSE FOR", order_detail.catalog, 1) > 0
  And Searchstr("QUANTITY", EOL.comment, 1) > 0
  And EOL.component_changed=False
  And EOL.assigned_to <> "IS"
Then EOL.suggested_action Is reassign_to_IS
  And EOL.explanation := "Potential system issue with ship loose items
quantity"
```

Rule 150

```
If (UpCase(EOL.reason) = "LIST ERROR" Or UpCase(EOL.reason) = "NOT ON
ASSEMBLY SHEET")
  And Searchstr("SHIP-LOOSE FOR", order_detail.catalog, 1) > 0
  And Searchstr("QUANTITY", EOL.comment, 1) > 0
  And EOL.component_changed=False
  And EOL.assigned_to = "IS"
Then EOL.suggested_action Is keep
  And EOL.explanation := "IS already knows about the ship loose item
problem"
```

Rule Aut1

```
If EOL.interior_material="BIRCH PAPER" And EOL.reason="NOT RECEIVED"
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.9714286 and importance=0.6381961"
```

Rule Aut2

If EOL.reason="NOT RECEIVED" And EOL.thickness=0.5
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.8785047 and importance=0.5950694"

Rule Aut3

If EOL.reason="NOT RECEIVED" And EOL.material_type_name="PLY"
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.8571429 and importance=0.6003343"

Rule Aut4

If EOL.reason="NOT RECEIVED" And EOL.location_name="D-21 OFFICE"
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.8523154 and importance=0.6619176"

Rule Aut5

If order_detail.material="BIRCH PAPER" And EOL.location_name="D-21
OFFICE"
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.7756757 and importance=0.5399498"

Rule Aut6

If order_detail.material="BIRCH PAPER"
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.7747989 and importance=0.5397786"

Rule Aut7

If EOL.reason="OUT OF SIZE SPEC" And EOL.location_name="D-21 OFFICE"
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.7721894 and importance=0.5335476"

Rule Aut8

If order_detail.material="BIRCH PAPER" And EOL.interior_material="BIRCH
PAPER"
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.7641791 and importance=0.5280317"

Rule Aut9

If EOL.material_type_name="PBC" And EOL.interior_material="BIRCH PAPER"
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.73297 and importance=0.5114006"

Rule Aut10

If EOL.interior_material="BIRCH PAPER" And
order_detail.division="LEGACY"
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with

probability=0.7272727 and importance=0.5387334"

Rule Aut11

If EOL.interior_material="BIRCH PAPER" And EOL.facility="LEGACY"

Then EOL.suggested_action Is reassign_to_D21

And EOL.explanation := "Item appear on current data with

probability=0.7150838 and importance=0.5427729"

Rule Aut12

If EOL.interior_material="BIRCH PAPER" And EOL.thickness=0.5

Then EOL.suggested_action Is reassign_to_D21

And EOL.explanation := "Item appear on current data with

probability=0.7132617 and importance=0.5583268"

Rule Aut13

If EOL.interior_material="BIRCH PAPER" And EOL.material_type_name="PLY"

Then EOL.suggested_action Is reassign_to_D21

And EOL.explanation := "Item appear on current data with

probability=0.7132075 and importance=0.5522459"

Rule Aut14

If EOL.interior_material="BIRCH PAPER" And EOL.location_name="D-21
OFFICE"

Then EOL.suggested_action Is reassign_to_D21

And EOL.explanation := "Item appear on current data with

probability=0.7119816 and importance=0.6354561"

Rule Aut15

If EOL.interior_material="BIRCH PAPER"

Then EOL.suggested_action Is reassign_to_D21

And EOL.explanation := "Item appear on current data with

probability=0.7118774 and importance=0.6359319"

Rule Aut16

If EOL.interior_material="BIRCH PAPER" And

order_detail.orderType="STANDARD"

Then EOL.suggested_action Is reassign_to_D21

And EOL.explanation := "Item appear on current data with

probability=0.7014925 and importance=0.5949845"

Rule Aut17

If EOL.interior_material="BIRCH PAPER" And EOL.quantity=1

Then EOL.suggested_action Is reassign_to_D21

And EOL.explanation := "Item appear on current data with

probability=0.7013453 and importance=0.5908064"

Rule Aut18

If EOL.interior_material="BIRCH PAPER" And order_detail.division="BATH"

Then EOL.suggested_action Is reassign_to_D21

And EOL.explanation := "Item appear on current data with

probability=0.6864111 and importance=0.502279"

Rule Aut19

If EOL.material_type_name="PLY" And EOL.facility="BATH"

Then EOL.suggested_action Is reassign_to_D21

And EOL.explanation := "Item appear on current data with

probability=0.667147 and importance=0.5010242"

Rule Aut20

If EOL.thickness=0.5 And EOL.material_type_name="PLY"
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.6670403 and importance=0.525677"

Rule Aut21

If EOL.thickness=0.5 And EOL.facility="LEGACY"
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.6622222 and importance=0.4947182"

Rule Aut22

If EOL.thickness=0.5 And order_detail.division="BATH"
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.6532567 and importance=0.4706051"

Rule Aut23

If EOL.thickness=0.5 And order_detail.division="LEGACY"
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.652819 and importance=0.4867851"

Rule Aut24

If EOL.facility="BATH" And EOL.location_name="D-21 OFFICE"
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.6516246 and importance=0.5405655"

Rule Aut25

If EOL.thickness=0.5 And EOL.location_name="D-21 OFFICE"
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.6471009 and importance=0.5802416"

Rule Aut26

If EOL.thickness=0.5 And EOL.quantity=1
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.6425159 and importance=0.5527119"

Rule Aut27

If EOL.interior_material="BIRCH" And EOL.facility="BATH"
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.6392157 and importance=0.4582245"

Rule Aut28

If EOL.thickness=0.5
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.6375176 and importance=0.5725164"

Rule Aut29

If order_detail.production_week=40 And EOL.location_name="D-21 OFFICE"
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.6334056 and importance=0.4487753"

Rule Aut30

If EOL.material_type_name="PLY" And order_detail.division="LEGACY"
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.6237624 and importance=0.46553"

Rule Aut31

If EOL.thickness=0.5 And order_detail.orderType="STANDARD"
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.6226102 and importance=0.5236762"

Rule Aut32

If EOL.material_type_name="PBC" And EOL.location_name="D-21 OFFICE"
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.6197802 and importance=0.4841999"

Rule Aut33

If EOL.interior_material="BIRCH" And order_detail.division="BATH"
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.6169491 and importance=0.4478268"

Rule Aut34

If EOL.material_type_name="PBC" And EOL.thickness=0.5
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.614 and importance=0.4368049"

Rule Aut35

If EOL.material_type_name="PLY" And EOL.thickness=0.75
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.6111111 and importance=0.4593825"

Rule Aut36

If order_detail.division="LEGACY" And EOL.location_name="D-21 OFFICE"
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.6041108 and importance=0.493256"

Rule Aut37

If EOL.thickness=0.75 And EOL.location_name="D-21 OFFICE"
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.6013384 and importance=0.4817218"

Rule Aut38

If EOL.material_type_name="MDF" And EOL.location_name="D-21 OFFICE"

Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.5961995 and importance=0.4151383"

Rule Aut39

If EOL.location_name="D-21 OFFICE" And order_detail.division="BATH"
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.594251 and importance=0.5825597"

Rule Aut40

If EOL.material_type_name="PBC"
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.5905759 and importance=0.4607593"

Rule Aut41

If EOL.interior_material="BIRCH" And EOL.thickness=0.75
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.5839874 and importance=0.4234908"

Rule Aut42

If EOL.material_type_name="PLY" And EOL.quantity=1
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.5838032 and importance=0.5926585"

Rule Aut43

If EOL.material_type_name="PLY" And EOL.location_name="D-21 OFFICE"
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.5830258 and importance=0.6356506"

Rule Aut44

If EOL.material_type_name="PLY"
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.5825287 and importance=0.6362997"

Rule Aut45

If EOL.material_type_name="PLY" And order_detail.division="BATH"
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.5802675 and importance=0.4767191"

Rule Aut46

If EOL.material_type_name="PBC" And order_detail.orderType="STANDARD"
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.5795314 and importance=0.435448"

Rule Aut47

If EOL.location_name="D-21 OFFICE"
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with

probability=0.5780521 and importance=1.919607"

Rule Aut48

If EOL.location_name="D-21 OFFICE" And EOL.quantity=1
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.5776173 and importance=0.930047"

Rule Aut49

If EOL.material_type_name="PLY" And order_detail.material="BIRCH"
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.5740741 and importance=0.4154837"

Rule Aut50

If EOL.material_type_name="PLY" And order_detail.orderType="STANDARD"
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.5737113 and importance=0.5746871"

Rule Aut51

If EOL.location_name="D-21 OFFICE" And
order_detail.orderType="STANDARD"
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.5680751 and importance=0.9787645"

Rule Aut52

If EOL.interior_material="BIRCH" And EOL.material_type_name="PLY"
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.5675287 and importance=0.4136042"

Rule Aut53

If EOL.facility="CUSTOM" And EOL.location_name="D-21 OFFICE"
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.5651214 and importance=0.3914631"

Rule Aut54

If EOL.material_type_name="PBC" And EOL.quantity=1
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.5557185 and importance=0.4012937"

Rule Aut55

If EOL.interior_material="BIRCH" And EOL.location_name="D-21 OFFICE"
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.5549683 and importance=0.4237031"

Rule Aut56

If EOL.facility="LEGACY" And EOL.location_name="D-21 OFFICE"
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.5531161 and importance=0.4701348"

Rule Aut57

If EOL.material_type_name="MDF"
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.5507559 and importance=0.3795698"

Rule Aut58

If EOL.facility="OELWEIN-MARKETPLACE" And EOL.location_name="D-21
OFFICE"
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.5491526 and importance=0.3877552"

Rule Aut59

If EOL.material_type_name="PLY" And EOL.facility="LEGACY"
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.5351812 and importance=0.4022811"

Rule Aut60

If EOL.reason="NOT RECEIVED" And EOL.quantity=1
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.530288 and importance=0.4029719"

Rule Aut61

If EOL.interior_material="BIRCH" And EOL.quantity=1
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.5258526 and importance=0.389973"

Rule Aut62

If order_detail.material="BIRCH" And EOL.location_name="D-21 OFFICE"
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.5220513 and importance=0.3912929"

Rule Aut63

If EOL.interior_material="BIRCH"
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.5208333 and importance=0.3926931"

Rule Aut64

If EOL.prod_group="MITERED" And EOL.location_name="F-10 EOL IN"
Then EOL.suggested_action Is reassign_to_F40
And EOL.explanation := "Item appear on current data with
probability=0.5207921 and importance=0.648921"

Rule Aut65

If EOL.reason="NOT RECEIVED" And EOL.facility="LEGACY"
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.52 and importance=0.3607468"

Rule Aut66

If EOL.prod_group="MITERED" And EOL.thickness=0.75
Then EOL.suggested_action Is reassign_to_F40
And EOL.explanation := "Item appear on current data with
probability=0.5198413 and importance=0.6477289"

Rule Aut67

If EOL.interior_material="BIRCH" And order_detail.orderType="STANDARD"
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.5194229 and importance=0.3824859"

Rule Aut68

If EOL.reason="NOT RECEIVED" And order_detail.division="LEGACY"
Then EOL.suggested_action Is reassign_to_D21
And EOL.explanation := "Item appear on current data with
probability=0.5147929 and importance=0.3494168"

End

APPENDIX B
METARULES SOURCE CODE

prjMetaRules/frmMain.frm

```

Attribute VB_Name = "frmMain"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
Option Explicit
'-----
Dim rs As New RSMetaRules
Dim QO As New QueryObject

Dim fileName As String
Dim fileToWrite As String
Dim currLine As String
Dim rule As Object
Dim decision As Object

'-----
' Procedure : Command1_Click
' Author    : XPMUser
' Date      : 05/1/2011
' Purpose   :
'-----
Private Sub Command1_Click()
    Dim astrSplitLines() As String
    Dim astrSplitItems() As String
    Dim ruleSplit() As String
    Dim rhsSplit() As String
    Dim i As Integer
    Dim linNum As Integer
    Dim incDec As String
    Dim incReason As String
    Dim incObject As String
    Dim incField As String
    Dim incOP As String
    Dim incValue As String
    Dim incType As String
    Dim count As Integer
    Dim outBuff As String
    Dim deptName() As String
    'Dim availFileNum As Integer
    Dim availFileToWrite As Integer
    Dim newAutRules As Integer

    fileName = App.Path + "\DMMRULES.TXT"
    fileToWrite = App.Path + "\NEWRULES.TXT"

    newAutRules = 0

    availFileToWrite = FreeFile

```

```

Open fileToWrite For Output As #availFileToWrite
Dim strInMEM As String

Dim fso As New FileSystemObject
Dim fil As File
Dim ts As TextStream
Dim rslt As Boolean
Dim fileLines() As String
Dim currProb As Single
Dim currImportance As Single

If fso.FileExists(fileName) Then
    Set fil = fso.GetFile(fileName)
    Set ts = fil.OpenAsTextStream(ForReading)
    strInMEM = ts.ReadAll
    ts.Close
Else
    MsgBox ("Input File Doesn't Exist")
    Exit Sub
End If

fileLines = Split(strInMEM, vbCrLf)

count = 1

For linNum = 0 To UBound(fileLines)
    astrSplitLines = Split(fileLines(linNum), "|")
    If (UBound(astrSplitLines) > 0) Then
        currLine = astrSplitLines(0)
        currProb = astrSplitLines(1)
        currImportance = astrSplitLines(2)
        astrSplitItems = Split(currLine, "->")
        If (UBound(astrSplitItems) > 0) Then
            deptName = Split(astrSplitItems(1), "=")
            ruleSplit = Split(astrSplitItems(0), ",")
            If (UBound(ruleSplit) > 0) Then
                outBuff = ""
                Dim logCond As Boolean
                logCond = True
                For i = 0 To UBound(ruleSplit)
                    If logCond Then
                        incDec = ""
                        incReason = ""
                        incObject = ""
                        incField = ""
                        incOP = ""
                        incValue = ""
                        incType = ""
                        Call SplitRule(ruleSplit(i), Trim(deptName(1)), incDec,
incReason, incObject, incField, incOP, incValue, incType)
                        logCond = logCond And (incDec = "accept")
                    If incDec = "accept" Then
                        If i > 0 Then outBuff = outBuff + " AND "
                        outBuff = outBuff + incObject + "." + incField + incOP
                    If incType = "isString" Then
                        outBuff = outBuff + """" + Trim(incValue) + """"
                    End If
                Next i
            End If
        End If
    End If
End For

```

```

        Else
            outBuff = outBuff + Trim(incValue)
        End If
    End If
End If
Next
If logCond Then
    newAutRules = newAutRules + 1
    outBuff = "Rule Aut" + CStr(newAutRules) + vbCrLf + "IF " +
outBuff + vbCrLf + "THEN EOL.suggested_action Is reassign_to_" +
Trim(deptName(1)) + vbCrLf
    outBuff = outBuff + "AND EOL.explanation := ""Item appear
on current data with probability=" & currProb & " and importance=" &
currImportance & """" & vbCrLf

    Else
        incDec = "reject"
    End If
Else
    incDec = ""
    incReason = ""
    incObject = ""
    incField = ""
    incOP = ""
    incValue = ""
    incType = ""
    Call SplitRule(ruleSplit(0), Trim(deptName(1)), incDec,
incReason, incObject, incField, incOP, incValue, incType)
    If incDec = "accept" Then
        newAutRules = newAutRules + 1
        outBuff = "Rule Aut" + CStr(newAutRules) + vbCrLf + "IF " +
incObject + "." + incField + incOP
        If incType = "isString" Then
            outBuff = outBuff + """" + Trim(incValue) + """"
        Else
            outBuff = outBuff + Trim(incValue)
        End If
        outBuff = outBuff + vbCrLf + "THEN EOL.suggested_action Is
reassign_to_" + Trim(deptName(1)) + vbCrLf
        outBuff = outBuff + "AND EOL.explanation := ""Item appear
on current data with probability=" & currProb & " and importance=" &
currImportance & """" & vbCrLf
    End If
End If
End If

    If incDec = "accept" Then
        Print #availFileToWrite, outBuff
    End If

    Labell.Caption = "Count : " + CStr(count) + vbCrLf + incDec +
vbCrLf + incReason
    count = count + 1
End If
Next 'numLin
'Loop

```

```

Close #availFileToWrite

Labell.Caption = "Done."

On Error GoTo 0
Exit Sub

Command1_Click_Error:

MsgBox "Error " & Err.Number & " (" & Err.Description & ") in
procedure Command1_Click of Form frmMain"
End Sub

Private Sub Form_Load()
    'rs.QueryObject = QO
    Set rule = rs.Classes.RuleToInclude
    Set decision = rs.Classes.decision
End Sub

Private Sub SplitRule(ByVal ruleSplit As String, ByVal deptName As
String, ByRef incDec As String, _
ByRef incReason As String, ByRef incObject As String, ByRef
incField As String, _
ByRef incOP As String, ByRef incValue As String, ByRef incType As
String)
    Dim rhsSplit() As String

    rhsSplit = Split(ruleSplit, "=")
    If UBound(rhsSplit) = 0 Then
        rhsSplit = Split(ruleSplit, ">=")
        If UBound(rhsSplit) = 0 Then
            rhsSplit = Split(ruleSplit, "<=")
            If UBound(rhsSplit) = 0 Then
                rhsSplit = Split(ruleSplit, ">")
                If UBound(rhsSplit) = 0 Then
                    rhsSplit = Split(ruleSplit, "<")
                    If UBound(rhsSplit) > 0 Then incOP = "<"
                End If
            Else
                incOP = "<="
            End If
        Else
            incOP = ">="
        End If
    Else
        incOP = "="
    End If
    If UBound(rhsSplit) > 0 Then

        incValue = rhsSplit(1)

        rs.Reset
        rs.DemonStrategy = 4
        rule.fieldName = Trim(rhsSplit(0))
        rule.deptName = deptName
        If IsNumeric(rhsSplit(1)) Then

```

```
        incType = "isNumeric"  
    ElseIf (Trim(rhsSplit(1)) = "True" Or Trim(rhsSplit(1)) = "False")  
Then  
        incType = "isSimple"  
    Else  
        incType = "isString"  
    End If  
  
    rule.fieldType = incType  
    rs.DemonStrategy = 3  
  
    incDec = decision.Value  
    incReason = decision.reason  
    incObject = decision.ObjectName  
    incField = decision.fieldName  
  
End If  
End Sub
```


prjMetaRules/QueryObject.cls

```

Attribute VB_Name = "QueryObject"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = True
Attribute VB_PredeclaredId = False
Attribute VB_Exposed = False
Option Explicit
Implements VRMS.Queries
Private m_ObjectName As String
Private m_AttributeName As String
Private m_AttributeClass As VRMS.Attribute
Private m_AttributeObject As Object
Private Const QueryCaption = "Default QueryObject - Enter value for:"

'=====
'One and Only Public function. Do not change name or parameters.
'Queries_Query(ByVal ObjectName As String, ByVal AttributeName As
String, ByVal AttributeClass As Object, ByVal AttributeObject As
Object)
'=====

Public Sub Queries_Query(ByVal ObjectName As String, ByVal
AttributeName As String, ByVal AttributeClass As Object, ByVal
AttributeObject As Object)

    m_ObjectName = ObjectName
    m_AttributeName = AttributeName
    Set m_AttributeClass = AttributeClass
    Set m_AttributeObject = AttributeObject

    MsgBox "Query For " & ObjectName & "." & AttributeName

    Set AttributeClass = Nothing
    Set AttributeObject = Nothing

    Set m_AttributeClass = Nothing
    Set m_AttributeObject = Nothing

End Sub

```

prjMetaRules/ RSMetaRules.Dsr

```

!=====
! CLASS Declarations

Class RuleToInclude
With fieldName String
With fieldType Compound isString, isNumeric, isSimple
With fieldValue String
With deptName String

Class Decision
With value Compound accept, reject Init reject
With reason String Init " "
With fieldName String Init " "
With objectName Compound EOL,cut_list,order_detail,
PO,notification,assembly_sheet,modification,formula, unset Init unset
!=====
! RULE & DEMON Declarations
!Attribute validDept Simple

!Demon try0
!If (RuleToInclude.deptName = "DDG" Or RuleToInclude.deptName = "RND"
Or RuleToInclude.deptName = "IS" Or RuleToInclude.deptName = "BDS" Or
RuleToInclude.deptName = "CDS" Or RuleToInclude.deptName = "LDS" Or
RuleToInclude.deptName = "D21")
!Then validDept := True
!Else validDept := False

Demon try1
If (RuleToInclude.fieldName = "ORDER NUMBER" And
RuleToInclude.fieldType Is isNumeric)
!And (validDept = True)
Then Decision.value Is accept
  And Decision.objectName Is order_detail
  And Decision.fieldName := "order_number"

Demon try2
If (RuleToInclude.fieldName = "ORDER NUMBER" And
(RuleToInclude.fieldType Is isSimple Or RuleToInclude.fieldType Is
isString))
!And (validDept = True)
Then Decision.value Is reject

Demon try3
If (RuleToInclude.fieldName = "MATERIAL NAME" And
RuleToInclude.fieldType Is isString)
!And (validDept = True)
Then Decision.value Is accept
  And Decision.objectName Is order_detail
  And Decision.fieldName := "material"

Demon try4
If (RuleToInclude.fieldName = "FACILITY NAME" And
RuleToInclude.fieldType Is isString)
!And (validDept = True)
Then Decision.value Is accept

```

```

And Decision.objectName Is EOL
And Decision.fieldName := "facility"

```

Demon try5

```

If (RuleToInclude.fieldName = "INTERIOR MATERIAL NAME" And
RuleToInclude.fieldType Is isString)
!And (validDept = True)
Then Decision.value Is accept
And Decision.objectName Is EOL
And Decision.fieldName := "interior_material"

```

Demon try6

```

If (RuleToInclude.fieldName = "THICKNESS" And RuleToInclude.fieldType
Is isNumeric)
!And (validDept = True)
Then Decision.value Is accept
And Decision.objectName Is EOL
And Decision.fieldName := "thickness"

```

Demon try7

```

If (RuleToInclude.fieldName = "C DIVISION NAME" And
RuleToInclude.fieldType Is isString)
!And (validDept = True)
Then Decision.value Is accept
And Decision.objectName Is order_detail
And Decision.fieldName := "division"

```

Demon try8

```

If (RuleToInclude.fieldName = "C REASON" And RuleToInclude.fieldType
Is isString)
!And (validDept = True)
Then Decision.value Is accept
And Decision.objectName Is eol
And Decision.fieldName := "reason"

```

Demon try9

```

If (RuleToInclude.fieldName = "MATERIAL TYPE NAME" And
RuleToInclude.fieldType Is isString)
!And (validDept = True)
Then Decision.value Is accept
And Decision.objectName Is eol
And Decision.fieldName := "material_type_name"

```

Demon try10

```

If (RuleToInclude.fieldName = "C LOCATION NAME" And
RuleToInclude.fieldType Is isString)
!And (validDept = True)
Then Decision.value Is accept
And Decision.objectName Is eol
And Decision.fieldName := "location_name"

```

Demon try11

```

If (RuleToInclude.fieldName = "C ORDER TYPE DESCRIPTION" And
RuleToInclude.fieldType Is isString)
!And (validDept = True)
Then Decision.value Is accept

```

```
And Decision.objectName Is order_detail
And Decision.fieldName := "orderType"
```

```
Demon try12
```

```
  If (RuleToInclude.fieldName = "N PRODUCTION WEEK" And
RuleToInclude.fieldType Is isNumeric)
!And (validDept = True)
Then Decision.value Is accept
  And Decision.objectName Is order_detail
  And Decision.fieldName := "production_week"
```

```
Demon try13
```

```
  If (RuleToInclude.fieldName = "N COMPONENT QUANTITY" And
RuleToInclude.fieldType Is isNumeric)
!And (validDept = True)
Then Decision.value Is accept
  And Decision.objectName Is eol
  And Decision.fieldName := "quantity"
```

```
Demon try14
```

```
  If (RuleToInclude.fieldName = "PRODUCTION GROUP" And
RuleToInclude.fieldType Is isString)
!And (validDept = True)
Then Decision.value Is accept
  And Decision.objectName Is eol
  And Decision.fieldName := "prod_group"
```

```
End
```

APPENDIX C
MODEL VIEWER AND RULE EXTRACT
SOURCE CODE


```

58 }
59 }
60 catch (System.Exception ex)
61 {
62     MessageBox.Show(ex.Message, "Connection Failed");
63 }
64 this.Cursor = Cursors.Default;
65 conn.Close();
66 }
67 private void comboModel1_SelectedIndexChanged(object sender, EventArgs e)
68 {
69     ShowModel(panel1, comboModel1.Text);
70 }
71 private void comboModel2_SelectedIndexChanged(object sender, EventArgs e)
72 {
73 }
74 /// <summary>
75 /// Show the selected model into the appropriate panel
76 /// </summary>
77 /// <param name="panel"></param>
78 /// <param name="modelName"></param>
79 private void ShowModel(Panel panel, string modelName)
80 {
81     AdomdConnection conn = new AdomdConnection();
82     try
83     {
84         MiningModelViewerControl viewer = null;
85         MiningModel model = null;
86         MiningService service = null;
87         // Clear any existing controls from the panel
88         if (panel.HasChildren)
89             panel.Controls.Clear();
90         // Connect to server
91         conn.ConnectionString = ConnectionString;
92         conn.Open();
93         // Determine the viewer type based on the model service and
94         // instantiate the correct viewer
95         model = conn.MiningModels[modelName];
96         service = conn.MiningServices[model.Algorithm];
97         if (service.ViewerType == "Microsoft_Cluster_Viewer")
98             viewer = new ClusterViewer();
99         else if (service.ViewerType == "Microsoft_99 Tree_Viewer")
100             viewer = new TreeViewer();
101         else if (service.ViewerType == "Microsoft_NaiveBayesian_Viewer")
102             viewer = new NaiveBayesViewer();
103         else if (service.ViewerType == "Microsoft_SequenceCluster_Viewer")
104             viewer = new SequenceClusterViewer();
105         else if (service.ViewerType == "Microsoft_TimeSeries_Viewer")
106             viewer = new TimeSeriesViewer();
107         else if (service.ViewerType == "Microsoft_AssociationRules_Viewer")
108             viewer = new AssociationViewer();
109         else if (service.ViewerType == "Microsoft_NeuralNetwork_Viewer")
110             viewer = new NeuralNetViewer();
111         else throw new System.Exception("Custom Viewers not supported");
112         // Set up and load the viewer
113         viewer.ConnectionString = ConnectionString;
114         viewer.MiningModelName = modelName;
115         viewer.Dock = DockStyle.Fill;
116         panel.Controls.Add(viewer);
117         viewer.LoadViewerData(null);
118     }
119     catch (System.Exception ex)

```

```

120 {
121     MessageBox.Show(ex.Message, "Model Load");
122 }
123 conn.Close();
124 }
125 private void button1_Click(object sender, EventArgs e)
126 {
127     string strTextFilePath = "DMMRules.txt";
128     AdomdConnection conn = new AdomdConnection(
129     string.Format("Data Source=localhost;Catalog={0}", m_Catalog));
130     conn.Open();
131     string commandText = "SELECT top 100 FROM " +
132     "{ SELECT " +
133     "NODE_CAPTION AS [Rule], " +
134     "NODE_PROBABILITY AS [Rule_Probability], " +
135     "MSOLAP_NODE_SCORE AS [Importance] " +
136     "FROM " +
137     "[{POND Department Association Model}.CONTENT " +
138     "WHERE NODE_TYPE=8) AS A " +
139     "WHERE " +
140     "Rule_Probability > 0.5 AND " +
141     "(VBA!InStr(Rule, ' = ->') = 0 AND " +
142     "VBA!InStr(Rule, ' = ,') = 0) " +
143     "ORDER BY [Rule_Probability] DESC ";
144     AdomdCommand cmd = new AdomdCommand(commandText, conn);
145     AdomdDataReader dr = cmd.ExecuteReader(CommandBehavior.CloseConnection);
146     // output the rows in the DataReader
147     using (StreamWriter sw = new StreamWriter(strTextFilePath))
148     {
149         while (dr.Read())
150         {
151             sw.WriteLine(dr.GetString(0));
152         }
153         dr.Close();
154     };
155     conn.Close();
156     MessageBox.Show("Rule analysis finished. See file for details", "Finished",
157     MessageBoxButtons.OK);
158 }
159 }
//-----
1 -
2 // <autogenerated>
3 // This code was generated by a tool.
4 // Runtime Version:2.0.40607.42
5 //
6 // Changes to this file may cause incorrect behavior and will be lost if
7 // the code is regenerated.
8 // </autogenerated>
9 //-----
---
10 namespace DataMiningViewerClient.Properties
11 {
12     public partial class Settings : System.Configuration.ApplicationSettingsBase
13     {
14         private static Settings m_Value;
15         private static object m_SyncObject = new object();
16         public static Settings Value
17         {
18             get

```



```

19 {
20 if ((Settings.m_Value == null))
21 {
22 System.Threading.Monitor.Enter(Settings.m_SyncObject);
23 if ((Settings.m_Value == null))
24 {
25 try
26 {
27 Settings.m_Value = new Settings();
28 }
29 finally
30 {
31 System.Threading.Monitor.Exit(Settings.m_SyncObject);
32 }
33 }
34 }
35 return Settings.m_Value;
36 }
37 }
38 }
39 }
namespace DataMiningViewerClient
{
3 partial class frmMain
4 {
5 /// <summary>
6 /// Required designer variable.
7 /// </summary>
8 private System.ComponentModel.IContainer components = null;
9 private System.Windows.Forms.Button btnConnect;
10 private System.Windows.Forms.ComboBox comboModel1;
11 private System.Windows.Forms.Label label1;
12 private System.Windows.Forms.Panel panel1;
13 private System.Windows.Forms.Button button1;
14 /// <summary>
15 /// Clean up any resources being used.
16 /// </summary>
17 protected override void Dispose(bool disposing)
18 {
19 if (disposing && (components != null))
20 {
21 components.Dispose();
22 }
23 base.Dispose(disposing);
24 }
25 /// <summary>
26 /// Required method for Designer support - do not modify
27 /// the contents of this method with the code editor.
28 /// </summary>
29 private void InitializeComponent()
30 {
31 this.btnConnect = new System.Windows.Forms.Button();
32 this.comboModel1 = new System.Windows.Forms.ComboBox();
33 this.label1 = new System.Windows.Forms.Label();
34 this.panel1 = new System.Windows.Forms.Panel();
35 this.button1 = new System.Windows.Forms.Button();
36 this.SuspendLayout();
37 //
38 // btnConnect
39 //
40 this.btnConnect.Location = new System.Drawing.Point(9, 13);
41 this.btnConnect.Name = "btnConnect";

```

```

40 this.btnConnect.Size = new System.Drawing.Size(125, 26);
43 this.btnConnect.TabIndex = 1;
44 this.btnConnect.Text = "&Connect";
45 this.btnConnect.Click += new System.EventHandler(this.btnConnect_Click);
46 //
47 // comboModell
48 //
49 this.comboModell.FormattingEnabled = true;
50 this.comboModell.Location = new System.Drawing.Point(196, 17);
51 this.comboModell.Name = "comboModell";
52 this.comboModell.Size = new System.Drawing.Size(237, 21);
this.comboModell.53 Sorted = true;
54 this.comboModell.TabIndex = 2;
this.comboModell.SelectedIndexChanged += new System.EventHandler(this.
comboModell_SelectedIndexChanged);
55
56 //
57 // labell
58 //
59 this.labell.AutoSize = true;
60 this.labell.Location = new System.Drawing.Point(151, 20);
61 this.labell.Name = "labell";
62 this.labell.Size = new System.Drawing.Size(39, 13);
63 this.labell.TabIndex = 3;
64 this.labell.Text = "Model:";
65 //
66 // panell
67 //
68 this.panell.Location = new System.Drawing.Point(9, 46);
69 this.panell.Name = "panell";
70 this.panell.Size = new System.Drawing.Size(765, 467);
71 this.panell.TabIndex = 4;
72 //
73 // button1
74 //
75 this.button1.Location = new System.Drawing.Point(699, 15);
76 this.button1.Name = "button1";
77 this.button1.Size = new System.Drawing.Size(75, 23);
78 this.button1.TabIndex = 5;
79 this.button1.Text = "Get Rules";
80 this.button1.UseVisualStyleBackColor = true;
81 this.button1.Click += new System.EventHandler(this.button1_Click);
82 //
83 // frmMain
84 //
85 this.ClientSize = new System.Drawing.Size(786, 525);
86 this.Controls.Add(this.button1);
87 this.Controls.Add(this.panell);
88 this.Controls.Add(this.labell);
89 this.Controls.Add(this.comboModell);
90 this.Controls.Add(this.btnConnect);
91 this.Name = "frmMain";
92 this.Text = "Model Compare";
93 this.ResumeLayout(false);
94 this.PerformLayout();
95 }
96 }
97 }
namespace DataMiningViewerClient
{
partial class dlgConnect
{

```

```

5 /// <summary>
6 /// Required designer variable.
7 /// </summary>
8 private System.ComponentModel.IContainer components = null;
9 private System.Windows.Forms.Button OK;
10 private System.Windows.Forms.Button btnCancel;
11 private System.Windows.Forms.Label label1;
12 private System.Windows.Forms.TextBox txtServer;
13 private System.Windows.Forms.ComboBox cmbCatalog;
14 private System.Windows.Forms.Label label2;
15 private System.Windows.Forms.Button btnTest;
16 /// <summary>
17 /// Clean up any resources being used.
18 /// </summary>
19 protected override void Dispose(bool disposing)
20 {
21 if (disposing && (components != null))
22 {
23 components.Dispose();
24 }
25 base.Dispose(disposing);
26 }
27 /// <summary>
28 /// Required method for Designer support - do not modify
29 /// the contents of this method with the code editor.
30 /// </summary>
31 private void InitializeComponent()
32 {
33 this.OK = new System.Windows.Forms.Button();
34 this.btnCancel = new System.Windows.Forms.Button();
35 this.label1 = new System.Windows.Forms.Label();
36 this.txtServer = new System.Windows.Forms.TextBox();
37 this.cmbCatalog = new System.Windows.Forms.ComboBox();
38 this.label2 = new System.Windows.Forms.Label();
39 this.btnTest = new System.Windows.Forms.Button();
40 this.SuspendLayout();
41 //
42 // OK
43 //
44 this.OK.Location = new System.Drawing.Point(86, 152);
45 this.OK.Name = "OK";
46 this.OK.TabIndex = 0;
47 this.OK.Text = "OK";
48 this.OK.Click += new System.EventHandler(this.OK_Click);
49 //
50 // btnCancel
51 //
52 this.btnCancel.DialogResult = System.Windows.Forms.DialogResult.Cancel;
53 this.btnCancel.Location = new System.Drawing.Point(168, 152);
54 this.btnCancel.Name = "btnCancel";
55 this.btnCancel.TabIndex = 1;
56 this.btnCancel.Text = "Cancel";
57 //
58 // label1
59 //
60 this.label1.Location = new System.Drawing.Point(4, 24);
61 this.label1.Name = "label1";
62 this.label1.Size = new System.Drawing.Size(47, 14);
63 this.label1.TabIndex = 0;
64 this.label1.Text = "&Server:";
65 //
66 // txtServer

```

```

67 //
68 this.txtServer.Location = new System.Drawing.Point(58, 21);
69 this.txtServer.Name = "txtServer";
70 this.txtServer.Size = new System.Drawing.Size(185, 20);
71 this.txtServer.TabIndex = 2;
72 this.txtServer.Leave += new System.EventHandler(this.txtServer_Leave);
this.txtServer.TextChanged += new System.EventHandler(this.
txtServer_TextChanged);
73
74 //
75 // cmbCatalog
76 //
77 this.cmbCatalog.FormattingEnabled = true;
78 this.cmbCatalog.Location = new System.Drawing.Point(58, 64);
79 this.cmbCatalog.Name = "cmbCatalog";
80 this.cmbCatalog.Size = new System.Drawing.Size(185, 21);
81 this.cmbCatalog.TabIndex = 3;
this.cmbCatalog.SelectedIndexChanged += new System.EventHandler(this.
cmbCatalog_SelectedIndexChanged);
82
this.cmbCatalog.TextChanged += new System.EventHandler(this.
cmbCatalog_TextChanged);
83
84 //
85 // label2
86 //
87 this.label2.AutoSize = true;
88 this.label2.Location = new System.Drawing.Point(4, 67);
89 this.label2.Name = "label2";
90 this.label2.Size = new System.Drawing.Size(43, 14);
91 this.label2.TabIndex = 4;
92 this.label2.Text = "&Catalog";
93 //
94 // btnTest
95 //
96 this.btnTest.Location = new System.Drawing.Point(168, 102);
97 this.btnTest.Name = "btnTest";
98 this.btnTest.TabIndex = 5;
99 this.btnTest.Text = "&Test";
100 this.btnTest.Click += new System.EventHandler(this.btnTest_Click);
101 //
102 // dlgConnect
103 //
104 this.AcceptButton = this.OK;
105 this.CancelButton = this.btnCancel;
106 this.ClientSize = new System.Drawing.Size(262, 189);
this.Controls.Add(this.btnTest);
108 this.Controls.Add(this.label2);
109 this.Controls.Add(this.cmbCatalog);
110 this.Controls.Add(this.txtServer);
111 this.Controls.Add(this.label1);
112 this.Controls.Add(this.btnCancel);
113 this.Controls.Add(this.OK);
114 this.Name = "dlgConnect";
115 this.Text = "Connect";
116 this.Load += new System.EventHandler(this.dlgConnect_Load);
117 this.ResumeLayout(false);
118 this.PerformLayout();
119 }
120 }
121 }

```

```

-----
1 -
2 // <autogenerated>
3 // This code was generated by a tool.
4 // Runtime Version:2.0.40607.42
5 //
6 // Changes to this file may cause incorrect behavior and will be lost if
7 // the code is regenerated.
8 // </autogenerated>
9 //-----
-----
10 namespace DataMiningViewerClient.Properties
11 {
12 using System;
13 using System.IO;
14 using System.Resources;
15 /// <summary>
16 /// A strongly-typed resource class, for looking up localized strings, etc.
17 /// </summary>
18 // This class was auto-generated by the Strongly Typed Resource Builder
19 // class via a tool like ResGen or Visual Studio.NET.
20 // To add or remove a member, edit your .ResX file then rerun ResGen
21 // with the /str option, or rebuild your VS project.
22 class Resources
23 {
24 private static System.Resources.ResourceManager _resMgr;
25 private static System.Globalization.CultureInfo _resCulture;
26 /// <summary>
27 /// Returns the cached ResourceManager instance used by this class.
28 /// </summary>
29 [System.ComponentModel.EditorBrowsableAttribute(System.ComponentModel.
EditorBrowsableState.Advanced)]
30 public static System.Resources.ResourceManager ResourceManager
31 {
32 get
33 {
34 if ((_resMgr == null))
35 {
36 System.Resources.ResourceManager temp = new System.Resources.
ResourceManager("Resources", typeof(Resources).Assembly);
37 _resMgr = temp;
38 }
39 return _resMgr;
40 }
41 }
42 /// <summary>
43 /// Overrides the current thread's CurrentUICulture property for all
44 /// resource lookups using this strongly typed resource class.
45 /// </summary>
46 [System.ComponentModel.EditorBrowsableAttribute(System.ComponentModel.
EditorBrowsableState.Advanced)]
47 public static System.Globalization.CultureInfo Culture
48 {
49 get
50 {
51 return _resCulture;
52 }
53 set
54 {

```

```

55 _resCulture = value;
56 }
57 }
58 /*FamANDAssem*
59 internal Resources()
60 {
61 }
62 }
63 }
//-----
1 -
2 // <copyright file="dlgConnect.cs" company="Microsoft">
3 // Copyright (c) Microsoft Corporation. All rights reserved.
4 // </copyright>
5 //-----
---
6 #region Using directives
7 using System;
8 using System.Collections.Generic;
9 using System.ComponentModel;
10 using System.Data;
11 using System.Drawing;
12 using System.Text;
13 using System.Windows.Forms;
14 using Microsoft.AnalysisServices.AdomdClient;
15 namespace DataMiningViewerClient
16 {
17 partial class dlgConnect : Form
18 {
19 public string m_ServerName;
20 private string m_CatalogName;
21 public string Server
22 {
23 get
24 {
25 return m_ServerName;
26 }
27 set
28 {
29 m_ServerName = value;
30 }
31 }
32 public string Catalog
33 {
34 get
35 {
36 return m_CatalogName;
37 }
38 set
39 {
40 m_CatalogName = value;
41 }
42 }
43 public string ConnectionString
44 {
45 get
46 {
47 return "Provider=MSOLAP.3;Data Source=" + m_ServerName + ";Initial Catalog="
+ m_CatalogName;
48 }
49 }

```

```

50 public dlgConnect()
51 {
52     m_CatalogName = "";
53     m_ServerName = "";
54     InitializeComponent();
55 }
56 private void dlgConnect_Load(object sender, EventArgs e)
57 {
58     txtServer.Text = m_ServerName;
59     LoadCatalog();
60     cmbCatalog.Text = m_CatalogName;
61 }
62 private void LoadCatalog()
63 {
64     AdomdConnection conn = null;
65     cmbCatalog.Items.Clear();
66     try
67     {
68         conn = new AdomdConnection(this.ConnectionString);
69         this.Cursor = Cursors.WaitCursor;
70         conn.Open();
71         DataSet ds = conn.GetSchemaDataSet(AdomdSchemaGuid.Catalogs, null);
72         foreach (DataRow row in ds.Tables[0].Rows)
73             cmbCatalog.Items.Add(row[0].ToString());
74     }
75     catch (System.Exception)
76     {
77         /**/
78     }
79     this.Cursor = Cursors.Default;
80     if (conn != null)
81         conn.Close();
82 }
83 private void txtServer_Leave(object sender, EventArgs e)
84 {
85     LoadCatalog();
86     if (cmbCatalog.Items.Count > 0)
87         cmbCatalog.SelectedIndex = 0;
88 }
89 private void cmbCatalog_SelectedIndexChanged(object sender, EventArgs e)
90 {
91     m_CatalogName = cmbCatalog.Text;
92 }
93 private void btnTest_Click(object sender, EventArgs e)
94 {
95     AdomdConnection conn = new AdomdConnection(this.ConnectionString);
96     try
97     {
98         conn.Open();
99         conn.Close();
100     MessageBox.Show("Test Succeeded", "Connection Succeeded");
101     }
102     catch (System.Exception ex)
103     {
104         MessageBox.Show(ex.Message, "Connection Failed");
105     }
106 }
107 private void txtServer_TextChanged(object sender, EventArgs e)
108 {
109     m_ServerName = txtServer.Text;
110 }
111 private void OK_Click(object sender, EventArgs e)

```

```

112 {
113 this.DialogResult = DialogResult.OK;
114 Close();
115 }
116 private void cmbCatalog_TextChanged(object sender, EventArgs e)
117 {
118 m_CatalogName = cmbCatalog.Text;
119 }
120 }
121 }
#region 1 Using directives
2 using System;
3 using System.Collections.Generic;
4 using System.Windows.Forms;
5 namespace DataMiningViewerClient
6 {
7 static class Program
8 {
9 /// <summary>
10 /// The main entry point for the application.
11 /// </summary>
12 static void Main()
13 {
14 Application.EnableVisualStyles();
15 Application.Run(new frmMain());
16 }
17 }
18 }
#region 1 Using directives
2 using System.Reflection;
3 using System.Runtime.CompilerServices;
4 #endregion
5 // General Information about an assembly is controlled through the following
6 // set of attributes. Change these attribute values to modify the information
7 // associated with an assembly.
8 [assembly: AssemblyTitle("DataMiningViewerClient")]
9 [assembly: AssemblyDescription("")]
10 [assembly: AssemblyConfiguration("")]
11 [assembly: AssemblyCompany("Microsoft Corporation")]
12 [assembly: AssemblyProduct("DataMiningViewerClient")]
13 [assembly: AssemblyCopyright("Copyright © Microsoft Corporation 2004")]
14 [assembly: AssemblyTrademark("")]
15 [assembly: AssemblyCulture("")]
16 // Version information for an assembly consists of the following four
17 // values:
18 // Major Version
19 // Minor Version
20 // Build Number
21 // Revision
22 //
23 // You can specify all the values or you can default the Revision and Build
24 // Numbers
25 // by using the '*' as shown below:
26 [assembly: AssemblyVersion("1.0.*")]

```


APPENDIX D
DMMRULES FILE CONTENT

INTERIOR MATERIAL NAME = BIRCH PAPER, C REASON = NOT RECEIVED -> PONC
DEPARTMENT = D21|0.971428571428571|0.638196085370721
C REASON = NOT RECEIVED, THICKNESS = 0.5 -> PONC DEPARTMENT =
D21|0.878504672897196|0.595069400283542
C REASON = NOT RECEIVED, MATERIAL TYPE NAME = PLY -> PONC DEPARTMENT =
D21|0.857142857142857|0.600334376021998
C REASON = NOT RECEIVED, C LOCATION NAME = D-21 OFFICE -> PONC DEPARTMENT =
D21|0.852315394242804|0.661917619111067
MATERIAL NAME = BIRCH PAPER, SUBASSEMBLY FLAG = Missing -> PONC DEPARTMENT =
D21|0.776881720430108|0.540990613166992
MATERIAL NAME = BIRCH PAPER, C LOCATION NAME = D-21 OFFICE -> PONC DEPARTMENT =
D21|0.775675675675676|0.539949844391506
MATERIAL NAME = BIRCH PAPER -> PONC DEPARTMENT =
D21|0.774798927613941|0.539778651356722
C REASON = OUT OF SIZE SPEC, C LOCATION NAME = D-21 OFFICE -> PONC DEPARTMENT =
D21|0.772189349112426|0.533547552563792
MATERIAL NAME = BIRCH PAPER, INTERIOR MATERIAL NAME = BIRCH PAPER -> PONC
DEPARTMENT = D21|0.764179104477612|0.528031719568459
INTERIOR MATERIAL NAME = BIRCH PAPER, GLAZE = NONE -> PONC DEPARTMENT =
D21|0.743073047858942|0.521882256716829
MATERIAL TYPE NAME = PBC, INTERIOR MATERIAL NAME = BIRCH PAPER -> PONC
DEPARTMENT = D21|0.732970027247956|0.511400644408801
INTERIOR MATERIAL NAME = BIRCH PAPER, C DIVISION NAME = LEGACY -> PONC
DEPARTMENT = D21|0.727272727272727|0.53873338107265
INTERIOR MATERIAL NAME = BIRCH PAPER, SUBASSEMBLY FLAG = Missing -> PONC
DEPARTMENT = D21|0.721995926680244|0.588225684057887
INTERIOR MATERIAL NAME = BIRCH PAPER, FACILITY NAME = LEGACY -> PONC DEPARTMENT
= D21|0.715083798882682|0.542772940384682
INTERIOR MATERIAL NAME = BIRCH PAPER, THICKNESS = 0.5 -> PONC DEPARTMENT =
D21|0.71326164874552|0.558326802083296
INTERIOR MATERIAL NAME = BIRCH PAPER, MATERIAL TYPE NAME = PLY -> PONC
DEPARTMENT = D21|0.713207547169811|0.552245886718871
INTERIOR MATERIAL NAME = BIRCH PAPER, C LOCATION NAME = D-21 OFFICE -> PONC
DEPARTMENT = D21|0.711981566820276|0.63545613153883
INTERIOR MATERIAL NAME = BIRCH PAPER -> PONC DEPARTMENT =
D21|0.711877394636015|0.635931886653854
INTERIOR MATERIAL NAME = BIRCH PAPER, C ORDER TYPE DESCRIPTION = STANDARD ->
PONC DEPARTMENT = D21|0.701492537313433|0.5949844630186
INTERIOR MATERIAL NAME = BIRCH PAPER, N COMPONENT QUANTITY = 1 -> PONC
DEPARTMENT = D21|0.701345291479821|0.590806353598634
THICKNESS = 0.5, GLAZE = NONE -> PONC DEPARTMENT =
D21|0.696280991735537|0.49924617963308
INTERIOR MATERIAL NAME = BIRCH PAPER, C DIVISION NAME = BATH -> PONC DEPARTMENT
= D21|0.686411149825784|0.502279022811977
MATERIAL TYPE NAME = PLY, FACILITY NAME = BATH -> PONC DEPARTMENT =
D21|0.667146974063401|0.501024174077321
THICKNESS = 0.5, MATERIAL TYPE NAME = PLY -> PONC DEPARTMENT =
D21|0.667040358744395|0.52567704414909
THICKNESS = 0.5, FACILITY NAME = LEGACY -> PONC DEPARTMENT =
D21|0.662222222222222|0.494718189816877
SUBASSEMBLY FLAG = Existing, C LOCATION NAME = D-21 OFFICE -> PONC DEPARTMENT =
D21|0.661596958174905|0.507777092760122
MATERIAL TYPE NAME = PLY, SUBASSEMBLY FLAG = Existing -> PONC DEPARTMENT =
D21|0.65585106382979|0.498134218299143
THICKNESS = 0.5, C DIVISION NAME = BATH -> PONC DEPARTMENT =
D21|0.653256704980843|0.47060507162917
THICKNESS = 0.5, C DIVISION NAME = LEGACY -> PONC DEPARTMENT =
D21|0.652818991097923|0.486785154671592
FACILITY NAME = BATH, C LOCATION NAME = D-21 OFFICE -> PONC DEPARTMENT =
D21|0.651624548736462|0.540565479362054
THICKNESS = 0.5, C LOCATION NAME = D-21 OFFICE -> PONC DEPARTMENT =
D21|0.647100930565497|0.580241599342314

THICKNESS = 0.5, N COMPONENT QUANTITY = 1 -> PONC DEPARTMENT =
 D21|0.642515923566879|0.552711885553662
 INTERIOR MATERIAL NAME = BIRCH, FACILITY NAME = BATH -> PONC DEPARTMENT =
 D21|0.63921568627451|0.458224503087891
 THICKNESS = 0.5 -> PONC DEPARTMENT = D21|0.637517630465444|0.572516386541346
 INTERIOR MATERIAL NAME = BIRCH, SUBASSEMBLY FLAG = Existing -> PONC DEPARTMENT
 = D21|0.636363636363636|0.446956603036622
 MATERIAL TYPE NAME = PLY, GLAZE = NONE -> PONC DEPARTMENT =
 D21|0.633776091081594|0.455548317602292
 N PRODUCTION WEEK = 40, C LOCATION NAME = D-21 OFFICE -> PONC DEPARTMENT =
 D21|0.633405639913232|0.448775331522364
 THICKNESS = 0.5, SUBASSEMBLY FLAG = Missing -> PONC DEPARTMENT =
 D21|0.627502175805048|0.521547406431578
 MATERIAL TYPE NAME = PLY, C DIVISION NAME = LEGACY -> PONC DEPARTMENT =
 D21|0.623762376237624|0.465530041949569
 THICKNESS = 0.5, C ORDER TYPE DESCRIPTION = STANDARD -> PONC DEPARTMENT =
 D21|0.622610141313383|0.523676193570468
 GLAZE = NONE, C LOCATION NAME = D-21 OFFICE -> PONC DEPARTMENT =
 D21|0.620775969962453|0.472710018245966
 MATERIAL TYPE NAME = PBC, C LOCATION NAME = D-21 OFFICE -> PONC DEPARTMENT =
 D21|0.61978021978022|0.484199915772155
 INTERIOR MATERIAL NAME = BIRCH, C DIVISION NAME = BATH -> PONC DEPARTMENT =
 D21|0.616949152542373|0.447826844209976
 MATERIAL TYPE NAME = PBC, THICKNESS = 0.5 -> PONC DEPARTMENT =
 D21|0.614|0.436804928519422
 MATERIAL TYPE NAME = PLY, THICKNESS = 0.75 -> PONC DEPARTMENT =
 D21|0.611111111111111|0.459382516949862
 C DIVISION NAME = LEGACY, C LOCATION NAME = D-21 OFFICE -> PONC DEPARTMENT =
 D21|0.604110813226095|0.493255973366058
 THICKNESS = 0.75, C LOCATION NAME = D-21 OFFICE -> PONC DEPARTMENT =
 D21|0.601338432122371|0.481721792337138
 MATERIAL TYPE NAME = MDF, C LOCATION NAME = D-21 OFFICE -> PONC DEPARTMENT =
 D21|0.596199524940618|0.415138328844598
 C LOCATION NAME = D-21 OFFICE, C DIVISION NAME = BATH -> PONC DEPARTMENT =
 D21|0.594250967385296|0.582559688080721
 MATERIAL TYPE NAME = PBC -> PONC DEPARTMENT =
 D21|0.590575916230367|0.460759308457224
 INTERIOR MATERIAL NAME = BIRCH, THICKNESS = 0.75 -> PONC DEPARTMENT =
 D21|0.583987441130298|0.423490756036932
 MATERIAL TYPE NAME = PLY, N COMPONENT QUANTITY = 1 -> PONC DEPARTMENT =
 D21|0.583803177857509|0.592658538006817
 MATERIAL TYPE NAME = PBC, SUBASSEMBLY FLAG = Missing -> PONC DEPARTMENT =
 D21|0.583243823845328|0.450946284855509
 MATERIAL TYPE NAME = PLY, C LOCATION NAME = D-21 OFFICE -> PONC DEPARTMENT =
 D21|0.583025830258303|0.635650547299088
 MATERIAL TYPE NAME = PLY -> PONC DEPARTMENT =
 D21|0.582528735632184|0.636299736642414
 MATERIAL TYPE NAME = PLY, C DIVISION NAME = BATH -> PONC DEPARTMENT =
 D21|0.580267558528428|0.476719079419046
 MATERIAL TYPE NAME = PBC, C ORDER TYPE DESCRIPTION = STANDARD -> PONC
 DEPARTMENT = D21|0.579531442663379|0.435448030505558
 C LOCATION NAME = D-21 OFFICE -> PONC DEPARTMENT =
 D21|0.578052126200274|1.91960743131366
 C LOCATION NAME = D-21 OFFICE, N COMPONENT QUANTITY = 1 -> PONC DEPARTMENT =
 D21|0.577617328519856|0.930046963541534
 MATERIAL TYPE NAME = PLY, MATERIAL NAME = BIRCH -> PONC DEPARTMENT =
 D21|0.574074074074074|0.41548366671955
 MATERIAL TYPE NAME = PLY, C ORDER TYPE DESCRIPTION = STANDARD -> PONC
 DEPARTMENT = D21|0.573711340206186|0.574687137855969
 C LOCATION NAME = D-21 OFFICE, C ORDER TYPE DESCRIPTION = STANDARD -> PONC
 DEPARTMENT = D21|0.568075117370892|0.978764486456572

INTERIOR MATERIAL NAME = BIRCH, MATERIAL TYPE NAME = PLY -> PONC DEPARTMENT =
D21|0.567528735632184|0.413604169127887
MATERIAL TYPE NAME = MDF, SUBASSEMBLY FLAG = Missing -> PONC DEPARTMENT =
D21|0.56651376146789|0.39134827015825
FACILITY NAME = CUSTOM, C LOCATION NAME = D-21 OFFICE -> PONC DEPARTMENT =
D21|0.565121412803532|0.391463139924073
MATERIAL TYPE NAME = PBC, N COMPONENT QUANTITY = 1 -> PONC DEPARTMENT =
D21|0.555718475073314|0.401293650679625
C LOCATION NAME = D-21 OFFICE, SUBASSEMBLY FLAG = Missing -> PONC DEPARTMENT =
D21|0.554971988795518|0.763383362637785
INTERIOR MATERIAL NAME = BIRCH, C LOCATION NAME = D-21 OFFICE -> PONC
DEPARTMENT = D21|0.554968287526427|0.423703060179839
FACILITY NAME = LEGACY, C LOCATION NAME = D-21 OFFICE -> PONC DEPARTMENT =
D21|0.553116147308782|0.470134837029673
MATERIAL TYPE NAME = MDF -> PONC DEPARTMENT =
D21|0.550755939524838|0.379569810774598
FACILITY NAME = OELWEIN-MARKETPLACE, C LOCATION NAME = D-21 OFFICE -> PONC
DEPARTMENT = D21|0.549152542372881|0.387755218480622
MATERIAL TYPE NAME = PLY, SUBASSEMBLY FLAG = Missing -> PONC DEPARTMENT =
D21|0.543921293042867|0.459978292022099
MATERIAL TYPE NAME = PLY, FACILITY NAME = LEGACY -> PONC DEPARTMENT =
D21|0.535181236673774|0.40228113006823
C REASON = NOT RECEIVED, N COMPONENT QUANTITY = 1 -> PONC DEPARTMENT =
D21|0.530287984111221|0.402971935320866
INTERIOR MATERIAL NAME = BIRCH, N COMPONENT QUANTITY = 1 -> PONC DEPARTMENT =
D21|0.525852585258526|0.389973031386808
MATERIAL NAME = BIRCH, C LOCATION NAME = D-21 OFFICE -> PONC DEPARTMENT =
D21|0.522051282051282|0.391292941929207
INTERIOR MATERIAL NAME = BIRCH -> PONC DEPARTMENT =
D21|0.520833333333333|0.392693088234327
PRODUCTION GROUP = MITERED, C LOCATION NAME = F-10 EOL IN -> PONC DEPARTMENT =
F40|0.520792079207921|0.648920953913159
C REASON = NOT RECEIVED, FACILITY NAME = LEGACY -> PONC DEPARTMENT =
D21|0.52|0.360746839439523
PRODUCTION GROUP = MITERED, THICKNESS = 0.75 -> PONC DEPARTMENT =
F40|0.51984126984127|0.647728838292659
INTERIOR MATERIAL NAME = BIRCH, C ORDER TYPE DESCRIPTION = STANDARD -> PONC
DEPARTMENT = D21|0.519422863485017|0.382485890662554
PRODUCTION GROUP = MITERED, SUBASSEMBLY FLAG = Existing -> PONC DEPARTMENT =
F40|0.517647058823529|0.646490093845384
C REASON = NOT RECEIVED, C DIVISION NAME = LEGACY -> PONC DEPARTMENT =
D21|0.514792899408284|0.349416769965023
C REASON = NOT RECEIVED, SUBASSEMBLY FLAG = Missing -> PONC DEPARTMENT =
D21|0.511373209772536|0.397095162540305

APPENDIX E
NEWRULES FILE CONTENT

Rule Aut1

```
IF EOL.interior_material="BIRCH PAPER" AND EOL.reason="NOT RECEIVED"  
THEN EOL.suggested_action Is reassign_to_D21  
AND EOL.explanation := "Item appear on current data with probability=0.9714286  
and importance=0.6381961"
```

Rule Aut2

```
IF EOL.reason="NOT RECEIVED" AND EOL.thickness=0.5  
THEN EOL.suggested_action Is reassign_to_D21  
AND EOL.explanation := "Item appear on current data with probability=0.8785047  
and importance=0.5950694"
```

Rule Aut3

```
IF EOL.reason="NOT RECEIVED" AND EOL.material_type_name="PLY"  
THEN EOL.suggested_action Is reassign_to_D21  
AND EOL.explanation := "Item appear on current data with probability=0.8571429  
and importance=0.6003343"
```

Rule Aut4

```
IF EOL.reason="NOT RECEIVED" AND EOL.location_name="D-21 OFFICE"  
THEN EOL.suggested_action Is reassign_to_D21  
AND EOL.explanation := "Item appear on current data with probability=0.8523154  
and importance=0.6619176"
```

Rule Aut5

```
IF order_detail.material="BIRCH PAPER" AND EOL.location_name="D-21 OFFICE"  
THEN EOL.suggested_action Is reassign_to_D21  
AND EOL.explanation := "Item appear on current data with probability=0.7756757  
and importance=0.5399498"
```

Rule Aut6

```
IF order_detail.material="BIRCH PAPER"  
THEN EOL.suggested_action Is reassign_to_D21  
AND EOL.explanation := "Item appear on current data with probability=0.7747989  
and importance=0.5397786"
```

Rule Aut7

```
IF EOL.reason="OUT OF SIZE SPEC" AND EOL.location_name="D-21 OFFICE"  
THEN EOL.suggested_action Is reassign_to_D21  
AND EOL.explanation := "Item appear on current data with probability=0.7721894  
and importance=0.5335476"
```

Rule Aut8

```
IF order_detail.material="BIRCH PAPER" AND EOL.interior_material="BIRCH PAPER"  
THEN EOL.suggested_action Is reassign_to_D21  
AND EOL.explanation := "Item appear on current data with probability=0.7641791  
and importance=0.5280317"
```

Rule Aut9

```
IF EOL.material_type_name="PBC" AND EOL.interior_material="BIRCH PAPER"  
THEN EOL.suggested_action Is reassign_to_D21  
AND EOL.explanation := "Item appear on current data with probability=0.73297  
and importance=0.5114006"
```

Rule Aut10

```
IF EOL.interior_material="BIRCH PAPER" AND order_detail.division="LEGACY"  
THEN EOL.suggested_action Is reassign_to_D21  
AND EOL.explanation := "Item appear on current data with probability=0.7272727  
and importance=0.5387334"
```

Rule Aut11

```
IF EOL.interior_material="BIRCH PAPER" AND EOL.facility="LEGACY"  
THEN EOL.suggested_action Is reassign_to_D21  
AND EOL.explanation := "Item appear on current data with probability=0.7150838  
and importance=0.5427729"
```

Rule Aut12

```
IF EOL.interior_material="BIRCH PAPER" AND EOL.thickness=0.5  
THEN EOL.suggested_action Is reassign_to_D21  
AND EOL.explanation := "Item appear on current data with probability=0.7132617  
and importance=0.5583268"
```

Rule Aut13

```
IF EOL.interior_material="BIRCH PAPER" AND EOL.material_type_name="PLY"  
THEN EOL.suggested_action Is reassign_to_D21  
AND EOL.explanation := "Item appear on current data with probability=0.7132075  
and importance=0.5522459"
```

Rule Aut14

```
IF EOL.interior_material="BIRCH PAPER" AND EOL.location_name="D-21 OFFICE"  
THEN EOL.suggested_action Is reassign_to_D21  
AND EOL.explanation := "Item appear on current data with probability=0.7119816  
and importance=0.6354561"
```

Rule Aut15

```
IF EOL.interior_material="BIRCH PAPER"  
THEN EOL.suggested_action Is reassign_to_D21  
AND EOL.explanation := "Item appear on current data with probability=0.7118774  
and importance=0.6359319"
```

Rule Aut16

```
IF EOL.interior_material="BIRCH PAPER" AND order_detail.orderType="STANDARD"  
THEN EOL.suggested_action Is reassign_to_D21  
AND EOL.explanation := "Item appear on current data with probability=0.7014925  
and importance=0.5949845"
```

Rule Aut17

```
IF EOL.interior_material="BIRCH PAPER" AND EOL.quantity=1  
THEN EOL.suggested_action Is reassign_to_D21  
AND EOL.explanation := "Item appear on current data with probability=0.7013453  
and importance=0.5908064"
```

Rule Aut18

```
IF EOL.interior_material="BIRCH PAPER" AND order_detail.division="BATH"  
THEN EOL.suggested_action Is reassign_to_D21  
AND EOL.explanation := "Item appear on current data with probability=0.6864111  
and importance=0.502279"
```

Rule Aut19

```
IF EOL.material_type_name="PLY" AND EOL.facility="BATH"  
THEN EOL.suggested_action Is reassign_to_D21  
AND EOL.explanation := "Item appear on current data with probability=0.667147  
and importance=0.5010242"
```

Rule Aut20

```
IF EOL.thickness=0.5 AND EOL.material_type_name="PLY"  
THEN EOL.suggested_action Is reassign_to_D21  
AND EOL.explanation := "Item appear on current data with probability=0.6670403  
and importance=0.525677"
```

Rule Aut21

```
IF EOL.thickness=0.5 AND EOL.facility="LEGACY"  
THEN EOL.suggested_action Is reassign_to_D21
```

AND EOL.explanation := "Item appear on current data with probability=0.6622222
and importance=0.4947182"

Rule Aut22

IF EOL.thickness=0.5 AND order_detail.division="BATH"
THEN EOL.suggested_action Is reassign_to_D21
AND EOL.explanation := "Item appear on current data with probability=0.6532567
and importance=0.4706051"

Rule Aut23

IF EOL.thickness=0.5 AND order_detail.division="LEGACY"
THEN EOL.suggested_action Is reassign_to_D21
AND EOL.explanation := "Item appear on current data with probability=0.652819
and importance=0.4867851"

Rule Aut24

IF EOL.facility="BATH" AND EOL.location_name="D-21 OFFICE"
THEN EOL.suggested_action Is reassign_to_D21
AND EOL.explanation := "Item appear on current data with probability=0.6516246
and importance=0.5405655"

Rule Aut25

IF EOL.thickness=0.5 AND EOL.location_name="D-21 OFFICE"
THEN EOL.suggested_action Is reassign_to_D21
AND EOL.explanation := "Item appear on current data with probability=0.6471009
and importance=0.5802416"

Rule Aut26

IF EOL.thickness=0.5 AND EOL.quantity=1
THEN EOL.suggested_action Is reassign_to_D21
AND EOL.explanation := "Item appear on current data with probability=0.6425159
and importance=0.5527119"

Rule Aut27

IF EOL.interior_material="BIRCH" AND EOL.facility="BATH"
THEN EOL.suggested_action Is reassign_to_D21
AND EOL.explanation := "Item appear on current data with probability=0.6392157
and importance=0.4582245"

Rule Aut28

IF EOL.thickness=0.5
THEN EOL.suggested_action Is reassign_to_D21
AND EOL.explanation := "Item appear on current data with probability=0.6375176
and importance=0.5725164"

Rule Aut29

IF order_detail.production_week=40 AND EOL.location_name="D-21 OFFICE"
THEN EOL.suggested_action Is reassign_to_D21
AND EOL.explanation := "Item appear on current data with probability=0.6334056
and importance=0.4487753"

Rule Aut30

IF EOL.material_type_name="PLY" AND order_detail.division="LEGACY"
THEN EOL.suggested_action Is reassign_to_D21
AND EOL.explanation := "Item appear on current data with probability=0.6237624
and importance=0.46553"

Rule Aut31

IF EOL.thickness=0.5 AND order_detail.orderType="STANDARD"
THEN EOL.suggested_action Is reassign_to_D21
AND EOL.explanation := "Item appear on current data with probability=0.6226102
and importance=0.5236762"

Rule Aut32

```
IF EOL.material_type_name="PBC" AND EOL.location_name="D-21 OFFICE"  
THEN EOL.suggested_action Is reassign_to_D21  
AND EOL.explanation := "Item appear on current data with probability=0.6197802  
and importance=0.4841999"
```

Rule Aut33

```
IF EOL.interior_material="BIRCH" AND order_detail.division="BATH"  
THEN EOL.suggested_action Is reassign_to_D21  
AND EOL.explanation := "Item appear on current data with probability=0.6169491  
and importance=0.4478268"
```

Rule Aut34

```
IF EOL.material_type_name="PBC" AND EOL.thickness=0.5  
THEN EOL.suggested_action Is reassign_to_D21  
AND EOL.explanation := "Item appear on current data with probability=0.614 and  
importance=0.4368049"
```

Rule Aut35

```
IF EOL.material_type_name="PLY" AND EOL.thickness=0.75  
THEN EOL.suggested_action Is reassign_to_D21  
AND EOL.explanation := "Item appear on current data with probability=0.6111111  
and importance=0.4593825"
```

Rule Aut36

```
IF order_detail.division="LEGACY" AND EOL.location_name="D-21 OFFICE"  
THEN EOL.suggested_action Is reassign_to_D21  
AND EOL.explanation := "Item appear on current data with probability=0.6041108  
and importance=0.493256"
```

Rule Aut37

```
IF EOL.thickness=0.75 AND EOL.location_name="D-21 OFFICE"  
THEN EOL.suggested_action Is reassign_to_D21  
AND EOL.explanation := "Item appear on current data with probability=0.6013384  
and importance=0.4817218"
```

Rule Aut38

```
IF EOL.material_type_name="MDF" AND EOL.location_name="D-21 OFFICE"  
THEN EOL.suggested_action Is reassign_to_D21  
AND EOL.explanation := "Item appear on current data with probability=0.5961995  
and importance=0.4151383"
```

Rule Aut39

```
IF EOL.location_name="D-21 OFFICE" AND order_detail.division="BATH"  
THEN EOL.suggested_action Is reassign_to_D21  
AND EOL.explanation := "Item appear on current data with probability=0.594251  
and importance=0.5825597"
```

Rule Aut40

```
IF EOL.material_type_name="PBC"  
THEN EOL.suggested_action Is reassign_to_D21  
AND EOL.explanation := "Item appear on current data with probability=0.5905759  
and importance=0.4607593"
```

Rule Aut41

```
IF EOL.interior_material="BIRCH" AND EOL.thickness=0.75  
THEN EOL.suggested_action Is reassign_to_D21  
AND EOL.explanation := "Item appear on current data with probability=0.5839874  
and importance=0.4234908"
```

Rule Aut42

```
IF EOL.material_type_name="PLY" AND EOL.quantity=1
THEN EOL.suggested_action Is reassign_to_D21
AND EOL.explanation := "Item appear on current data with probability=0.5838032
and importance=0.5926585"
```

Rule Aut43

```
IF EOL.material_type_name="PLY" AND EOL.location_name="D-21 OFFICE"
THEN EOL.suggested_action Is reassign_to_D21
AND EOL.explanation := "Item appear on current data with probability=0.5830258
and importance=0.6356506"
```

Rule Aut44

```
IF EOL.material_type_name="PLY"
THEN EOL.suggested_action Is reassign_to_D21
AND EOL.explanation := "Item appear on current data with probability=0.5825287
and importance=0.6362997"
```

Rule Aut45

```
IF EOL.material_type_name="PLY" AND order_detail.division="BATH"
THEN EOL.suggested_action Is reassign_to_D21
AND EOL.explanation := "Item appear on current data with probability=0.5802675
and importance=0.4767191"
```

Rule Aut46

```
IF EOL.material_type_name="PBC" AND order_detail.orderType="STANDARD"
THEN EOL.suggested_action Is reassign_to_D21
AND EOL.explanation := "Item appear on current data with probability=0.5795314
and importance=0.435448"
```

Rule Aut47

```
IF EOL.location_name="D-21 OFFICE"
THEN EOL.suggested_action Is reassign_to_D21
AND EOL.explanation := "Item appear on current data with probability=0.5780521
and importance=1.919607"
```

Rule Aut48

```
IF EOL.location_name="D-21 OFFICE" AND EOL.quantity=1
THEN EOL.suggested_action Is reassign_to_D21
AND EOL.explanation := "Item appear on current data with probability=0.5776173
and importance=0.930047"
```

Rule Aut49

```
IF EOL.material_type_name="PLY" AND order_detail.material="BIRCH"
THEN EOL.suggested_action Is reassign_to_D21
AND EOL.explanation := "Item appear on current data with probability=0.5740741
and importance=0.4154837"
```

Rule Aut50

```
IF EOL.material_type_name="PLY" AND order_detail.orderType="STANDARD"
THEN EOL.suggested_action Is reassign_to_D21
AND EOL.explanation := "Item appear on current data with probability=0.5737113
and importance=0.5746871"
```

Rule Aut51

```
IF EOL.location_name="D-21 OFFICE" AND order_detail.orderType="STANDARD"
THEN EOL.suggested_action Is reassign_to_D21
AND EOL.explanation := "Item appear on current data with probability=0.5680751
and importance=0.9787645"
```

Rule Aut52

```
IF EOL.interior_material="BIRCH" AND EOL.material_type_name="PLY"
THEN EOL.suggested_action Is reassign_to_D21
```

AND EOL.explanation := "Item appear on current data with probability=0.5675287
and importance=0.4136042"

Rule Aut53

IF EOL.facility="CUSTOM" AND EOL.location_name="D-21 OFFICE"
THEN EOL.suggested_action Is reassign_to_D21
AND EOL.explanation := "Item appear on current data with probability=0.5651214
and importance=0.3914631"

Rule Aut54

IF EOL.material_type_name="PBC" AND EOL.quantity=1
THEN EOL.suggested_action Is reassign_to_D21
AND EOL.explanation := "Item appear on current data with probability=0.5557185
and importance=0.4012937"

Rule Aut55

IF EOL.interior_material="BIRCH" AND EOL.location_name="D-21 OFFICE"
THEN EOL.suggested_action Is reassign_to_D21
AND EOL.explanation := "Item appear on current data with probability=0.5549683
and importance=0.4237031"

Rule Aut56

IF EOL.facility="LEGACY" AND EOL.location_name="D-21 OFFICE"
THEN EOL.suggested_action Is reassign_to_D21
AND EOL.explanation := "Item appear on current data with probability=0.5531161
and importance=0.4701348"

Rule Aut57

IF EOL.material_type_name="MDF"
THEN EOL.suggested_action Is reassign_to_D21
AND EOL.explanation := "Item appear on current data with probability=0.5507559
and importance=0.3795698"

Rule Aut58

IF EOL.facility="OELWEIN-MARKETPLACE" AND EOL.location_name="D-21 OFFICE"
THEN EOL.suggested_action Is reassign_to_D21
AND EOL.explanation := "Item appear on current data with probability=0.5491526
and importance=0.3877552"

Rule Aut59

IF EOL.material_type_name="PLY" AND EOL.facility="LEGACY"
THEN EOL.suggested_action Is reassign_to_D21
AND EOL.explanation := "Item appear on current data with probability=0.5351812
and importance=0.4022811"

Rule Aut60

IF EOL.reason="NOT RECEIVED" AND EOL.quantity=1
THEN EOL.suggested_action Is reassign_to_D21
AND EOL.explanation := "Item appear on current data with probability=0.530288
and importance=0.4029719"

Rule Aut61

IF EOL.interior_material="BIRCH" AND EOL.quantity=1
THEN EOL.suggested_action Is reassign_to_D21
AND EOL.explanation := "Item appear on current data with probability=0.5258526
and importance=0.389973"

Rule Aut62

IF order_detail.material="BIRCH" AND EOL.location_name="D-21 OFFICE"
THEN EOL.suggested_action Is reassign_to_D21
AND EOL.explanation := "Item appear on current data with probability=0.5220513
and importance=0.3912929"

Rule Aut63

```
IF EOL.interior_material="BIRCH"  
THEN EOL.suggested_action Is reassign_to_D21  
AND EOL.explanation := "Item appear on current data with probability=0.5208333  
and importance=0.3926931"
```

Rule Aut64

```
IF EOL.prod_group="MITERED" AND EOL.location_name="F-10 EOL IN"  
THEN EOL.suggested_action Is reassign_to_F40  
AND EOL.explanation := "Item appear on current data with probability=0.5207921  
and importance=0.648921"
```

Rule Aut65

```
IF EOL.reason="NOT RECEIVED" AND EOL.facility="LEGACY"  
THEN EOL.suggested_action Is reassign_to_D21  
AND EOL.explanation := "Item appear on current data with probability=0.52 and  
importance=0.3607468"
```

Rule Aut66

```
IF EOL.prod_group="MITERED" AND EOL.thickness=0.75  
THEN EOL.suggested_action Is reassign_to_F40  
AND EOL.explanation := "Item appear on current data with probability=0.5198413  
and importance=0.6477289"
```

Rule Aut67

```
IF EOL.interior_material="BIRCH" AND order_detail.orderType="STANDARD"  
THEN EOL.suggested_action Is reassign_to_D21  
AND EOL.explanation := "Item appear on current data with probability=0.5194229  
and importance=0.3824859"
```

Rule Aut68

```
IF EOL.reason="NOT RECEIVED" AND order_detail.division="LEGACY"  
THEN EOL.suggested_action Is reassign_to_D21  
AND EOL.explanation := "Item appear on current data with probability=0.5147929  
and importance=0.3494168"
```

APPENDIX F

50 CASES AND PONC EXPERT RESPONSE

PONC case	PONC Response	Description	Initial Dept	S1	S2	S1	S2
1	DDG	FaceFrame subassembly sheets were not including Perimeter Frames. However, this EOL should be reported as a miscommunication between Dealer Services, Plant and IS.	IS	VH	VH	DDG	DDG
2	F20	Item was produced correctly for this recessed medicine cabinet freestyle. Was shipped to Rough Mill @ Finish Mill and scanned by Mike Mentel @ 10/5/2011 9:25 AM.	IS	VL	VL	N/A	N/A
3	CDS	Incorrect size 7.3125 was entered in parts screen. From the drawing, it shows 7.75 for the return	IS	SL	SL	DDG	DDG
4	DDG	Item was reprocessed multiple times in an attempt to fix a formula problem by Toby Beddard, as reported in notifications	IS	VL	VL	N/A	N/A
5	IS	System was assigning incorrect item count to components. Only one item was being produced	IS	VL	SH	N/A	D21
6	LDS	Processor entered wrong species in custom order specs for a legacy plus order #2535363	IS	VL	SH	N/A	F40
7	LDS	Processor entered wrong species in custom order specs for a legacy plus order #2535363	IS	VL	SH	N/A	F40
8	IS	System was assigning incorrect item count to components. Only one item was being produced	IS	VH	VH	IS	IS
9	IS	See 408739	IS	VH	VH	IS	IS
10	M50	Plant reordered this component wrong. It was created correctly in the first place.	IS	VL	SH	N/A	D21
11	M50	See 408588	IS	VL	SH	N/A	D21
12	IS	System was assigning incorrect item count to components. Only one item was being produced. This issue was addressed and a fix was released	IS	VL	SH	N/A	D21
13	IS	See 409338	IS	VL	SH	N/A	D21
14	BDS	Order was deleted and IS was asked to reinstate it with initial PO. However, list was already printed and no cutlist was printed for these components.	IS	VL	SH	N/A	D21
15	PLY MILL	ON SELCO FP LIST & FM JIT LISTS	DDG	VL	VH	N/A	D21
16	PLY MILL	ON WOOD HOOD SELCO ANGULAR LIST & FM JIT LISTS	DDG	VL	VH	N/A	D21
17	L50	EMAILED TIM RICKERT HAVE NOT HEARD BACK	DDG	VL	SH	N/A	D21
18	L50	EMAILED TIM RICKERT HAVE NOT HEARD BACK	DDG	VL	SH	N/A	D21
19	PLY MILL	ON GABIANNI PENNISULA LIST & FM JIT LISTS	DDG	VL	VH	N/A	D21
20	PLY MILL	ON GABIANNI PENNISULA LIST & FM JIT LISTS	DDG	VL	VH	N/A	D21
21	LDS	PROCESSOR ERROR & MOD. ERROR, PER NOTIFICATION	DDG	VL	SH	N/A	D21
22	IS	PARTS COMING OUT WITH DIFFERENT WOOD SPECIES	DDG	VL	VL	N/A	N/A
PONC case	PONC Response	Description	Initial Dept	S1	S2	S1	S2

23	IS	PER NOTIFICATION	DDG	H	H	RND	RND
24	R&D	DRAWING CAME OUT AFTER PAPERWORK PRINTED	DDG	VL	SH	N/A	D21
25	LDS	PROCESSOR ERROR, COMMENTS DO NOT CHANGE CUTLIST	DDG	VL	SH	N/A	D21
26	LDS	CHANGE NOTIFICATION FROM MARY NOBIS, CTSO SENT 10/27	DDG	VL	SH	N/A	D21
27	LDS	CHANGE NOTIFICATION FROM MARY NOBIS, CTSO SENT 10/27	DDG	VL	SH	N/A	D21
28	LDS	CHANGE NOTIFICATION FROM MARY NOBIS, CTSO SENT 10/27	DDG	VL	SH	N/A	D21
29	LDS	CHANGE NOTIFICATION FROM MARY NOBIS, CTSO SENT 10/27	DDG	VL	SH	N/A	D21
30	LDS	CHANGE NOTIFICATION FROM MARY NOBIS, CTSO SENT 10/27	DDG	VL	SH	N/A	D21
31	R&D	NOTIFICATION FROM NORTHEY, ADJUSTED STOCK FORMULA	DDG	VL	L	N/A	N/A
32	R&D	NOTIFICATION FROM NORTHEY, ADJUSTED STOCK FORMULA	DDG	VL	L	N/A	N/A
33	R&D	PER NOTIFICATION FROM JEFF NORTHEY	DDG	VL	L	N/A	N/A
34	R&D	PER NOTIFICATION FROM JEFF NORTHEY	DDG	VL	L	N/A	N/A
35	R&D	PER NOTIFICATION FROM JEFF NORTHEY	DDG	VL	L	N/A	N/A
36	R&D	PER NOTIFICATION FROM JEFF NORTHEY	DDG	VL	L	N/A	N/A
37	R&D	PER NOTIFICATION FROM JEFF NORTHEY	DDG	VL	L	N/A	N/A
38	R&D	PER NOTIFICATION FROM JEFF NORTHEY	DDG	VL	L	N/A	N/A
39	MDS	CHANGE NOTIFICATION, CTSO SENT 10/31	DDG	VL	SH	N/A	D21
40	R&D	PER NOTIFICATION FROM JEFF NORTHEY	DDG	VL	L	N/A	N/A
41	R&D	PER NOTIFICATION FROM JEFF NORTHEY	DDG	VL	L	N/A	N/A
42	BDS	CHANGE NOTIFICATION, CTSO SENT 10/26	DDG	VH	VH	DDG	DDG
43	BDS	CHANGE NOTIFICATION, CTSO SENT 10/26	DDG	VH	VH	DDG	DDG
44	FINISH MILL	NO CHANGES, NO OTHER NOTIFICATIONS	DDG	VL	L	N/A	N/A
45	FINISH MILL	NO CHANGES, NO OTHER NOTIFICATIONS	DDG	VL	L	N/A	N/A
46	DDG	PATTY FIXED BY PUTTING IN TYPE NAME	DDG	VL	SH	N/A	D21
47	DDG	PATTY FIXED BY PUTTING IN TYPE NAME	DDG	VL	SH	N/A	D21
48	IS	PARTS COMING OUT WITH DIFFERENT WOOD SPECIES	DDG	VL	SH	N/A	D21
49	IS	PARTS COMING OUT WITH DIFFERENT WOOD SPECIES	DDG	VL	SH	N/A	D21
50	IS	PARTS COMING OUT WITH DIFFERENT WOOD SPECIES	DDG	VL	SH	N/A	D21