

Loughborough University Institutional Repository

An application of a rule-based system towards resolving exception events of business processes within a small Virtual Organization

This item was submitted to Loughborough University's Institutional Repository by the/an author.

Citation: DAS, B.P. ... et al, 2010. An application of a rule-based system towards resolving exception events of business processes within a small Virtual Organization. IN: Proceedings of 20th International Conference on Flexible Automation and Intelligent Manufacturing 2010 (FAIM 2010), California State University, USA, 12-14 July, paper 1053, pp. 1-8.

Additional Information:

- This is a conference paper.

Metadata Record: <https://dspace.lboro.ac.uk/2134/9796>

Version: Accepted for publication

Publisher: Curran Associates, Inc. (© FAIM 2010)

Please cite the published version.

This item was submitted to Loughborough's Institutional Repository (<https://dspace.lboro.ac.uk/>) by the author and is made available under the following Creative Commons Licence conditions.



CC creative commons
COMMONS DEED

Attribution-NonCommercial-NoDerivs 2.5

You are free:

- to copy, distribute, display, and perform the work

Under the following conditions:

BY: **Attribution.** You must attribute the work in the manner specified by the author or licensor.

Noncommercial. You may not use this work for commercial purposes.

No Derivative Works. You may not alter, transform, or build upon this work.

- For any reuse or distribution, you must make clear to others the license terms of this work.
- Any of these conditions can be waived if you get permission from the copyright holder.

Your fair use and other rights are in no way affected by the above.

This is a human-readable summary of the [Legal Code \(the full license\)](#).

[Disclaimer](#) 

For the full text of this licence, please go to:
<http://creativecommons.org/licenses/by-nc-nd/2.5/>

An application of a rule-based system towards resolving exception events of business processes within a small Virtual Organization

B. P. Das^{1*}, A. Shah¹, J. A. Harding¹, C. Palmer¹, R. Swarnkar¹, R. I. Young¹, and K. Popplewell²

¹Wolfson School Mechanical and Manufacturing Engineering, Loughborough University, Loughborough, LE 11 3TU, UK.

²Future Manufacturing Applied Research Centre, Coventry University, Coventry, CV 1 5FB, UK.

ABSTRACT

The motivation behind this research work is explained first within the context of a European Union funded project called "SYNERGY". A business scenario of a small virtual organization is presented to discuss what happens if exception events occur within the life cycle of a business scenario, and how they could be resolved to enable the business to operate successfully. To investigate these aspects, the paper briefly explores the role of events within the context of "Collaboration Moderator Services" which aim to increase awareness between collaborative partners during the workflow of a business scenario and then suggests a possible remedy through the use of a rule-based system. A description of a methodology adopted for creation and capture of knowledge needed to resolve the exception events, transform the captured knowledge into rule-sets and implement them in a commercial rule-based system – "XpertRule" is given. Screen shots of this implementation are provided in the paper. Based on this implementation, the paper then concludes with a discussion.

1. INTRODUCTION

The SYNERGY research project [1] aims to support small to medium sized networked enterprises (SMNE) by providing an infrastructure and Web-based services to enable them to collaborate and work together more easily and cost-effectively. These services would enable SMNEs to register, discover, capture, plan and run complex knowledge-based collaborative projects through formation of a virtual organization (VO). A VO has been defined in [2] as: "A temporary alliance of independent enterprises that come together to share skills, core competencies and resources in order to better respond to business opportunities and whose cooperation is supported by computer networks". The collaboration moderator services (CMS) is one of the service modules of the overall SYNERGY system, and is being developed by Loughborough University (LU), UK. This research paper relates to some of the work being carried out within the context of the CMS. An objective of CMS is to increase awareness and understanding between collaborating SMNEs by raising awareness of changes to "Objects of Interest" which might affect them. "Objects of Interest" considered in this paper are: New business opportunities; New collaboration opportunities or partners; Lessons learnt; Shortfalls in competencies, etc. All of these functionalities require some form of decision making activities. Therefore it is appropriate to explore the applicability of a rule-based system within the context of the CMS work package.

This research paper explores the suitability of a rule-based system in managing the "exception events" in business processes for many reasons. First it is an important approach to supporting any type of business organization, be it a single enterprise or a virtual organization consisting of two or more enterprises. For example in a business process, a number of pre-planned events, as well as unforeseen events, may occur. Within the context of this paper, an event is a part of the work-flow of an organisation. It may occur in a process or activity and may relate to many subsequent sub-processes or sub-activities. Similarly, all the unforeseen events are termed as "exception events". Under normal circumstances, if all the processes of a business perform their activities correctly and no exception event occurs, then the process should lead to satisfactory conclusion within a set time. However, this does not always happen. An exception event may occur while running a process, additionally a user may have to trigger a change in an activity plan or the strategy due to an unavoidable circumstances, which may affect others. These

* Corresponding author: E-mail: B.P.Das@lboro.ac.uk

changes could also be considered as exception events. The occurrence of these exception events is not desirable but as said earlier may happen. Generally they may be flagged-up, by some one responsible within the business process (for example by project participants, project leader, project manager etc.) or by the system itself, whenever a problem arises. In that situation, that particular “exception event” would require immediate attention and a satisfactory solution, so that the process or the event could return back to its original state where problem arose and subsequently the work could move-on. This clearly suggests that these eventualities need to be considered while designing the system and ideally appropriate solutions as well as methodology has to be provided by the system, so that satisfactory conclusion of the process can take place. The second reason for exploring rule-based systems was to explore the applicability of this concept within the context of one of the functional requirements of the CMS, which is “Monitoring and alerting partners to changes to object of interest which might affect them”.

Therefore to assess some of these aspects, in section two, a business scenario of a newly formed VO directed towards identifying a candidate pharmaceutical drug is presented, including typical list of processes that may take place within the organization. Following this in section three, an analysis of processes towards identifying related exception events is presented. The purpose of this section was to highlight the possible list of exception events that may happen under each process and explore the implications of these “exception events” on the business scenario. Section four, then discusses how the rule-sets are developed for these exception events to enable making decisions for possible solutions. Following the development of rule-sets, section five, then deals with the implementation aspects of these rule-sets within a commercially available rule-based system environment – “XpertRule”. Finally section six ends with a discussion based on this implementation.

2. VO BUSINESS SCENARIO

2.1 THE STORY LINE

A small pharmaceutical company (Company A) has been informed of a new business opportunity within a regionally based business environment for exploring and developing a new candidate drug. The competencies needed to service this business opportunity are found to be: ***business experience*** in developing new drugs, ***expertise*** in ***conducting experimental methods*** and ***expertise*** in ***computational design*** of new effective drugs. After considering the technical competency requirements, Company A has decided to form collaboration partnership with two or more organizations which have complimentary competencies to exploit that opportunity.

Company A, has identified two small organizations having the right expertise and has formed a temporary VO alliance after detailed discussions on confidentiality etc. and has agreed to lead and manage the project. The first organization happens to be an innovative academic laboratory (Company B) with expertise in experimental methods and the second organization happens to be a specialist computer laboratory (Company C) with expertise in computational design for new drugs.

This story line describes a situation where a number of small companies collaborate to solve a drug design problem as an early stage step in a drug discovery situation. This hypothetical story line will now be used to explore the suitability of the rule-based system in managing the “exception events” that may occur in various processes of this VO business. In order to assess this aspect, the next sub-section provides a list of processes that may take place in this business story line, which forms an important part of this assessment.

2.2 BUSINESS PROCESS LIST

After discussions amongst the business partners (Company A, Company B and Company C), an agreed list of processes needed to complete the project was finalized. Table 1 below shows a section of the process list (column 2). There are many major processes and each process may have number of sub-processes but they are not all shown here. Also the linkages between the individual processes should not be assumed to be sequential only, some of these processes may happen in parallel and in between these processes some decisions have to be taken thereby introducing constraints to those processes, which are not discussed here. The next logical step of this assessment is to analyse the processes in detail and identify possible exception events that may possibly occur under these processes, and this is the subject of the next section.

Table 1: A section of the processes list, exception events as identified for the VO

Column 1 No	Column 2 Process description	Column 3 Exception events
P1	Submit project proposal as requested	Unable to satisfy requirements with available resources.
P2	Respond to observations made on proposal	Unable to complete responses on observations within deadline
P3	Create VO & Support infrastructure	Unable to implement Support infrastructure within deadline
P4	Arrange Kick-off meeting	Partners unable to attend meeting
-----	-----	-----
P10	Run computer modelling activity	Computational methods fail
P11	Discuss computer modelling result	Can not agree on next steps on modelling
-----	-----	-----

3. EXCEPTION EVENT IDENTIFICATION

There are many exception events for each of the business processes in this scenario. Exception events are identified by posing a set of questions to Company personnel associated with these processes such as: “What can change”, “What problems may arise” etc. A section of the result of this analysis is presented in Table 1 (column 3). Ideally all the processes, sub-processes should be analyzed in-advance so that possible list of exception events are identified and possible preventive measures in the form of solutions (monitoring and managing) could be developed to ensure that the project could be completed satisfactorily. This leads to the next section, where solutions in the form of rule-sets for some of these exception events are discussed.

4. RULE-SETS FOR EXCEPTION EVENTS

Developing rule-sets for a decision or a solution is a two stage process. First it involves creating knowledge in the form of solutions and then interpreting this knowledge into rule-sets to enable a solution or decision to be reached based on certain conditions. Once these rule-sets are developed then they could be used for making decisions whenever these types of situations arise, possibly by implementing them in a suitable computing environment. The next two sub-sections briefly describe how such knowledge was created and then the rule-sets were developed from this knowledge for various exception events.

4.1 KNOWLEDGE CREATION

In this knowledge creation process, first, all the “exception events” that are identified for this business scenario are collated, then analyzed and then solutions are derived for each exception event by posing questions such as: What are the possible causes of occurrence of these exception events? What are the possible solutions or actions needed to cure the causes of these exception events? A section of the solution developed in this manner is presented in Table 2.

Table 2: A section of list of exception events and corresponding solutions for Process step P11

Activity number	Exception event	Possible solution when an exception event occurs
P11.1	Meeting organizer is unavailable	This exception event might have been generated by the participant who originally requested the Meeting organizer (MO) to set-up a meeting for discussing some important results. This participant expected a reply from the MO within a set date or within a set number of days. Because the participant did not receive a reply, so this message or exception event was generated by that participant and most probably sent out to the MO again and other partners. The possible options to respond to this exception event are: The originator of this event choose an alternative meeting organizer from the VO knowledge pool matching the competency required to do the role and request him to perform the subsequent activities. The other option will be for the originator to consider, if the meeting could be delayed, then wait for the availability of MO and when MO is available send a reminder possibly by email or by some other means of communication to arrange this delayed meeting for discussing the experimental result.
P11.2	Meeting organizer unable to set agenda alone	This exception event might have been generated by the MO himself. Possibly he does not have the knowledge or experience of setting up an agenda or not familiar with the subject matter of the meeting or the topics of the meeting. Under this situation the MO should contact the participant who requested for the meeting and ask for his advice so that the <i>objective</i> of the meeting could be captured and then complete the agenda with the participant's cooperation and then perform the next or subsequent activity that is circulate the agenda to relevant participants.
P11.3	Profile unavailable	Possibly this exception event has been generated by the MO himself. May be, the MO was trying to find out the competency of a particular partner from the partner's profile databases and was not successful in viewing the profile for some reasons. Under this circumstance, the MO should request (by email/or other communication means) the participant or the system manager who maintains this profile to cure the problem and wait until the problem is cured. Once the problem is resolved then MO may perform the next or subsequent activity as he planned.
-----	-----	-----

4.2 INTERPRETING KNOWLEDGE AS RULE SETS

In this activity, the solutions (knowledge) of each of these exception events as derived in the previous subsection, is translated into a list of rule-sets following the format of "IF a situation ..THEN an action ..". An example of one such rule-set is shown in figure 1. In this example, the exception event is "Meeting organizer (MO) is unavailable". After studying the text based solution, the meaning of this solution is then fitted into the "If .. and.. Then.." format in such a manner that it leads to the solution/solutions as expressed in the text. In this example there are five rule-sets and corresponding sub-branches, which are closely linked. This leads to the next section which involves implementation and validation of these rule-sets in an expert system software environment.

5. IMPLEMENTING RULE-SETS IN A COMPUTING ENVIRONMENT

This section briefly provides an overview of the XpertRule software system environment used for implementing the rule-sets, followed by the method used in implementing rule-sets derived in previous section and associated outputs (screen-shots) of this implementation. The XpertRule knowledge based environment is only one of several commercial knowledge based environments available in the market and is not preferred above others. It has been used because it was already available to the researchers undertaking this study, and has a good graphical interface enabling the rules to be clearly shown. The detailed discussions on rule-based systems is out of the scope of this paper. They are available in [3,4,5,6,7,8,9,10,11].

5.1 OVERVIEW OF XPERTRULE SYSTEM

The XpertRule system is an expert system environment, marketed by XpertRule Software Limited, enabling the creation of knowledge based applications, incorporating rules, expertise, know-how, procedures etc. for various

business purposes. XpertRule Knowledge Builder is available commercially in two editions: (a) Standard/Education edition (b) Enterprise edition. Further information is available in [12]. For this research paper, XpertRule Knowledge Builder (Standard/Education edition) has been used and the rules are represented through “Decision trees”. A decision tree uses a tree-like structure to relate decisions to their possible consequences. Figure 2, shows an example of a decision tree. A decision can be related to the “IF” part of a rule and its consequence to the “THEN” part of the rule. In XpertRule Knowledge Builder decisions are comprised of attributes (shown as hexagonal boxes) containing values (displayed as rectangular boxes). The consequence of a decision is displayed as a rounded rectangular box. For example, figure 2, shows a decision comprised of the attribute “SupplementaryMechanicalManufacturer” with a value “can be found” and the resulting consequence “increase production”.

5.2 RULE-SET IMPLEMENTATION

The XpertRule Knowledge Builder environment provides the option of representing rules through a decision tree, where a number of rule-sets are implemented in the form of knowledge modules as detailed in Table 3. These knowledge modules are then tested for validation using the “Run” command of the system, by selecting each knowledge module in turn. The “Run” command of the system makes a link between the knowledge module selected by the user and the inference engine of the system. The inference engine comes to a decision based on an interactive “question and answer session” with the user and produces a report. Two example implementations are briefly explained below.

Analysis of Exception Events for Process step P11
Relating to InnovationWell – Collaborative Drug Discovery VQ workflow.

Rule Set for P11.1
Meeting Organiser (MO) is unavailable.

<p>RuleSet_1</p> <p>IF alternative means of communication with MO are available THEN try RuleSet_1_Sub_Branch_1.</p> <p>IF there are no alternative means of communications with MO THEN try RuleSet_2</p>	<p>RuleSet_1_Sub_Branch_1</p> <p>IF alternative means of communications with MO yield desired result THEN take appropriate action</p> <p>IF MO cannot be reached by alternative means OR alternative means of communications do not yield desired results THEN try RuleSet_2</p>
<p>RuleSet_2</p> <p>IF there is an Assistant MO or a Deputised MO available for communication THEN try RuleSet_2_Sub_Branch_1.</p> <p>IF there is NO Assistant MO or Deputised MO available for communication THEN try RuleSet_3</p>	<p>RuleSet_2_Sub_Branch_1</p> <p>IF communication with Assistant or Deputy MO is successful THEN take appropriate action.</p> <p>IF communication with Assistant or Deputy MO is unsuccessful OR it fails to produce a solution THEN try RuleSet_3</p>
<p>RuleSet_3</p> <p>IF it is possible to postpone meeting until the MO is available THEN try RuleSet_3_Sub_Branch_1.</p> <p>IF it is NOT possible to postpone meeting until MO is available THEN try RuleSet_4</p>	<p>RuleSet_3_Sub_Branch_1</p> <p>IF VO is strategically willing to postpone meeting until the MO is available THEN postpone meeting.</p> <p>IF VO is NOT strategically willing to postpone meeting until the MO is available THEN try RuleSet_4</p>
<p>RuleSet_4</p> <p>IF it is possible to find an alternative MO from the VO knowledge pool that matches the competencies required THEN try RuleSet_4_Sub_Branch_1.</p> <p>IF it is NOT possible to find an alternative qualified MO THEN try RuleSet_5</p>	<p>RuleSet_4_Sub_Branch_1</p> <p>IF alternative MO from the VO knowledge pool matching competencies required is found THEN proceed with meeting.</p> <p>IF alternative qualified MO is unavailable THEN try RuleSet_5</p>
<p>RuleSet_5</p> <p>IF it is possible to find a broker to organise meeting THEN try RuleSet_4_Sub_Branch_1.</p> <p>IF it is NOT possible to find a broker to organise meeting THEN indefinitely postpone meeting.</p>	<p>RuleSet_5_Sub_Branch_1</p> <p>IF broker is found to organise meeting THEN take appropriate action.</p> <p>IF it is NOT possible to find a broker to organise meeting THEN indefinitely postpone meeting.</p>

Figure 1: An example of a rule-set P11.1

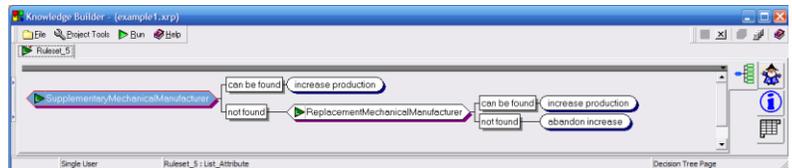


Figure 2: Screen shot of a decision tree

Table 3: A typical list of knowledge modules as implemented in XpertRule Knowledge builder environment

Knowledge module	Exception event	Rule-set number
1	Recipient responds on inability to attend.	Rule set P11_6
2	Meeting organiser (MO) is unavailable.	Rule set P11_1
3	Meeting organiser unable to set agenda alone.	Rule set P11_2
4	Calendar availability information not up to date.	Rule set P11_4_C
5	Meeting scheduler (MS) is unavailable.	Rule set P11_5_A
6	Participants cannot agree on Agenda.	Rule set P11_11_D
-----	-----	-----

IMPLEMENTATION 1: PARTICIPANTS CANNOT AGREE ON AGENDA

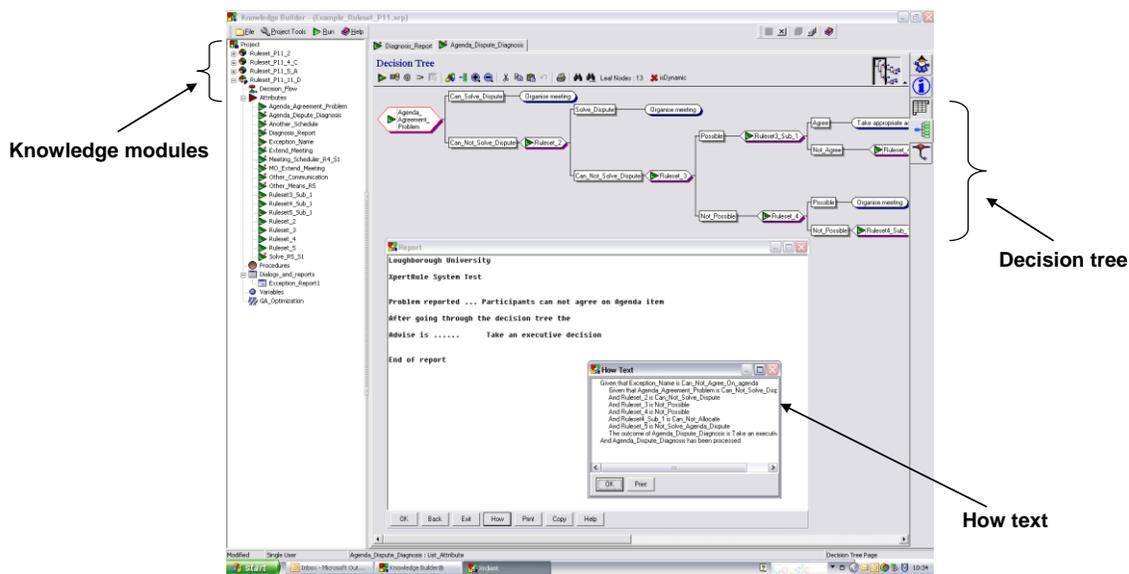


Figure 3: Decision output of an exception event - “Participants cannot agree on agenda”.

A screen shot of the output of implementation of the knowledge module relating to the exception event – “participants can not agree on agenda” is shown in figure 3. In this example, the user first opens the XpertRule Knowledge builder environment and then selects a project. Under this project folder, a list of knowledge modules would be displayed. For example Rule set P11_2, Rule set P11_4_C, Rule set P11_5_A, Rule set P11_11_D etc. In this case, the user selects the knowledge module “Rule set P11_11_D”, which relates to the exception event – “Participants cannot agree on Agenda”. The user then invokes the “Run” command of the XpertRule knowledge builder system, to find out a solution or advise. Following this an interactive conversation session takes place between the user and the system. Once a conclusion is reached by the inference engine of the system, a report will be displayed on the screen as shown in figure 3. The user could also view “how” this decision has been reached by the inference engine by pressing the “how” button of the report screen as shown in figure 3. For reason of clarity, a section of the decision tree as implemented within the system for this exception event is also shown in figure 3.

IMPLEMENTATION 2: EXCEPTION EVENT ANALYSIS

The idea behind the development of this knowledge module was to enable a user to analyse the effect of an exception event on other partners of the project. This is done by developing simple “rule-sets” to show the list of exception events, the possible nodes (partners) being affected by these exception events and the possible corresponding simple solutions with limited interaction between the user and the system. These possible simple solutions could be the synopsis of the detailed solutions discovered earlier for each exception event. One such example is shown in figure 4 below and briefly explained here.

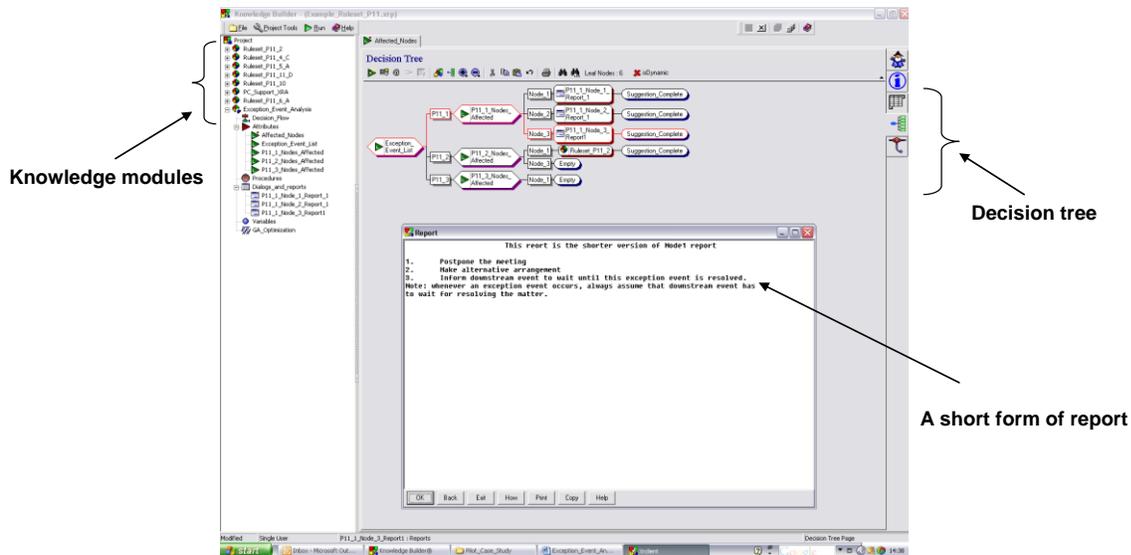


Figure 4: An example of a possible exception event and possible list of nodes that could be effected by the exception event and a possible report (solution) for that particular node.

Figure 4, shows the screen shot of the implementation of one of the knowledge module relating to implementation “Exception event analysis”. When the user selects this knowledge module from the project folder of the system, and then initiates the “Run” command of the XpertRule knowledge builder system, a list of exception events would be shown. For example: Meeting organizer unavailable, Meeting organizer unable to set agenda, Profile unavailable etc. A user may chose an exception event from this list, in this example, the user selected “P11_1:Meeting organiser unavailable”. Once this exception event is chosen by the user, the system would show the possible list of nodes (partners) that might be affected by this event as shown in figure 4. These nodes are basically the project partners. For example node 1 would be Company A, node 2 would be Company B and so on. This exception event affects all the three nodes (Node1, Node 2 and Node 3). The user may then choose a particular effected node for the possible solution. The inference engine of the system would then display the solution in the form of a report on the screen as shown in figure 4. For reason of clarity, the decision tree as implemented within the system for this knowledge module is also shown in figure 4.

5.3 OBSERVATIONS

These two examples demonstrate that solutions for exception events in the form of rule-sets can be implemented as decision trees in the XpertRule knowledge builder system and they can be stored as a set of knowledge modules. Later on, such knowledge modules could be consulted for solutions whenever confronted with such exception events within the context of the business scenario. Therefore developing such a system is useful and can assist user in resolving exception events or similar types of problems promptly.

The two examples shown here are slightly different. The first example shows the list of exception events in the form of knowledge modules and provides a solution in the form of report depending on the knowledge module chosen by user. The second example goes beyond that boundary. It enables user to view the effect of an exception event on participating partners of the VO and suggests possible solutions.

6. DISCUSSIONS

This research paper has presented through computer implementation, a methodology for testing the applicability of a rule based system in resolving exception events that may occur in a business process and concludes that such a system can be used to assist a user to resolve these issues.

The method developed involves: Analysis of the business scenario, identifying the list of processes of the business scenario, identifying list of exception events for each process, developing solutions for each of exception event, translating these solutions into rule-sets, implementing these rule-sets as decision trees in a rule-based system (in this case XpertRule Knowledge builder) and using these decision trees in the form of knowledge modules for solutions, when such exception events occur. Although the methodology has been developed within the context of a small pharmaceutical VO, they should be equally applicable to other business sectors, for example manufacturing, construction, chemicals processing, retailing as well as in supply-chain environment etc.

Implementation of various rule sets within the XpertRule system environment revealed that, there may be more than one solution for a particular exception event which would depend on set of conditions. Such conditional analysis for a solution could go to several levels of reasoning, for example up to 3-5 levels in some cases. It is easy to represent such conditional rules through a decision tree and instant visualization of such tree within the XpertRule system environment is found to be very helpful in understanding the situation.

Implementation of rule-sets for a solution through decision trees may sometimes be *time-consuming*, especially when there are many branches and sub-branches. It is easy to overlook and make mistakes in a complex decision tree. Therefore such trees should be kept short (possibly no more than 3 or 4 level of reasoning) to reduce errors. This may also enhance the responsiveness of the system. Solutions through rule-based system generally require interaction between user and the system, therefore due consideration (particularly responsiveness of the system and user's acceptance) should be given. Further research is required to explore the feasibility of integrating rule-based applications with a possible mainstream application system.

REFERENCES

- [1] <http://www.synergy-ist.eu/> (accessed April 19, 2010).
- [2] H. Luczak and A. Hauser: "Knowledge management in virtual organizations", *Proceedings of ICSSSM'05, International Conference on Services Systems and Services Management*, pp. 898, 2005.
- [3] A. A. Hopgood, *Knowledge-based systems for Engineers and Scientists*, CRC Press, Inc, Boca Raton, Fla, USA, 1993.
- [4] R. J. Mockler, *Knowledge-based systems for management decisions*, Prentice Hall, Englewood Cliffs, New Jersey, USA, 1989.
- [5] B. Coppin, *Artificial Intelligence Illuminated*, Jones and Barlett Publishers, Sudbury, Massachusetts, USA, 2004.
- [6] W. Wen, W. K. Wang and T. H. Wang: "A hybrid knowledge-based decision support system for enterprise mergers and acquisitions", *Expert Systems with Applications*, vol. 28, pp. 569-582, 2005.
- [7] E. Turban and J. E. Aronson, *Decision Support Systems and Intelligent Systems*, Sixth editn, Prentice Hall, New Jersey, USA, 2001.
- [8] M. W. Firebaugh, *Artificial Intelligence – A Knowledge-Based Approach*, Boyd & Fraser Publishing Company, Boston, USA, 1988.
- [9] P. H. Winston, *Artificial Intelligence*, 3rd Edition editn, Addison-Wesley Publishing Company, Reading, Massachusetts, USA, 1992.
- [10] J. S. Edwards, *Building Knowledge-Based Systems*, Pitman Publishing, London, UK, 1991.
- [11] M. Minsky, "A framework for representing knowledge" in *The Psychology of Computer Vision*, ed. P. Winston, McGraw-Hill, USA, pp. 211-277, 1975.
- [12] <http://www.XpertRule.com/> (accessed April 19, 2010).