An Evaluation Study of the RECOCASE Viewpoint Methodology

Debbie Richards and Anne Britt Fure

Computing Department,
Division of Information and Communication Sciences,
Macquarie University, Australia
richards@ics.mq.edu.au

We have developed a viewpoint methodology that captures use case descriptions from multiple viewpoints and automatically generates a visualisation of the individual and shared viewpoints to assist identification and resolution of conflicts. Our evaluation studies showed that our methodology and tool were practical and assisted the development of more complete use case descriptions which were representative of each member of the group. This paper describes the studies that were conducted.

1 Introduction

The capture of requirements from multiple viewpoints has been presented as a way to develop a more complete, consistent and representative set of requirements. It can also assist communication, future modification, conflict identification and resolution [2, 3]. Our approach, RECOCASE, assists the capture and RECOnciliation of functional requirements in the form of use case descriptions from multiple viewpoints. In keeping with the visual nature of other parts of the Unified Modelling Language (UML) object-oriented system development approach, we model the use case descriptions in a diagram to identify and reconcile differences. The RECOCASE methodology includes use case description guidelines, a controlled language to support natural language translation, a requirements engineering process model, a group decision support approach and a Computer Aided Software Engineering (CASE) tool. In this paper we present findings of our empirical investigation of the usability and usefulness of the RECOCASE methodology and tool. Our approach is described in detail in [4].

2. The RECOCASE methodology and tool: An Introduction

The RECOCASE methodology includes the following six iterative phases:

1. Requirements acquisition

The first phase starts by a group, led by the group facilitator/leader (GF), brainstorming a set of use cases. Viewpoints for each use case and viewpoint representative are identified. Asynchronously, viewpoint representatives enter use case descriptions in natural language into the RECOCASE tool. Natural language can however be ambiguous and complex in syntax and semantics. A set of writing

guidelines and controlled language is provided [6], but since users rarely follow guidelines, the tool can also detect and notify the viewpoint agent of problematic words, e.g. pronouns, modal verbs, and, or and negations.

2. Requirements translation,

After the viewpoint representatives have completed their descriptions, the tool will automatically parse the sentences into noun and verb phrases by using natural language techniques [5].

3. Concept generation,

Formal Concept Analysis (FCA) [7] is more commonly used for data analysis tasks to find structure and display relationships between concepts which consist of attributes and objects. We use FCA to generate and structure concepts. These concepts are used in the next step to create a visual representation of the use case descriptions.

4. Concept comparison and conflict detection,

The GF creates the diagrams by selecting the sentences he/she wants to be included. Different strategies for combining use case descriptions can be used. If there are few sentences, maybe one or two diagrams constructed by selecting sentences based on system interaction and actors are adequate. However, if there are a larger number of sentences (>25) the GF can choose to create diagrams based on logical units of entences. The GF will then be able to have a mental overview of the logical flow in the use case. The group members and the GF can use the line diagram to compare viewpoints and detect conflicts. The nodes in the graph represent the concepts, and in our diagrams these are words and phrases from the use case description. The nodes belonging to the same sentence are connected with lines. Sentences containing identical words or synonyms will share nodes.

5. Negotiation and Reconciliation

During the reconciliation process, the group and GF will look at the diagrams together to build a shared use case. Our negotiation strategies are described in [4].

6. Evaluation.

We use graph theory on the lattices to evaluate if the different viewpoints have become more similar, or if a new round of negotiation is necessary.

In the next section we will describe the study. Section 4 gives the results. Section 5 includes our summarised findings. The paper concludes with a synopsis of the results and our approach in general.

The Automated Video System (AVS) is being designed to allow customers to self check out videos, search for particular video titles, request new videos to be stocked by each store, and to make reservations for videos currently unavailable.

Each customer has his/her own customer card which is issued by members of the staff. Each card has a credit value and the customers can give money to staff members who then increase the card value. The customers use this card when they rent videos, and the price for renting a movie is subtracted from the credit value on the card. If the credit value is too low or the customer has any overdue films, the customer will not be allowed to rent a new movie. Any overdue fines must be paid to a staff member.

Figure 1. The Test case problem presented to participants in Test 1, 2 and 4.

3. Design of the study

The data presented in this paper are collected from surveys, rich text from observing the participants during the individual and group sessions, and examination of the use case descriptions. While a more empirical study could have been conducted with practitioners, we wanted to do initial studies first and the next best alternative was 3rd year computing students who all had some knowledge about use cases from current or previous studies and who would soon be in the workplace. We completed four separate tests, each time with three students. While the use of students is often doubtful, these studies were pilots that were conducted to identify weaknesses before usage on RE practitioners whose availability is much more limited. As these students can expect to be in industry in the following year and many in industry are less familiar with use cases we felt that students were not such an inferior substitute.

In tests 1 and 2 the test participants used the RECOCASE methodology and tool to create an Automated Video System (AVS) as described in Figure1. We created this problem as it was a novel application which combined two familiar situations: video rental via store staff and automated borrowing of items at a library. Test 3 was conducted with the same students as in Test 1, but this time they were given a new problem to solve. We organized the tests in this way to evaluate our methodology on first time and more experienced users. Test 4 was a separate test where three test participants solved the AVS problem without using RECOCASE.

The second author acted as the group facilitator (GF) in all tests, and assisted the participants in drawing the use case diagram, using the guidelines and the tool. A pilot study performed in June 2002 showed that the GF needs experience as a leader and with the tool or the test would be invalidated. As the approach was new to the second author it was a realistic test of what could be achieved with limited experience.

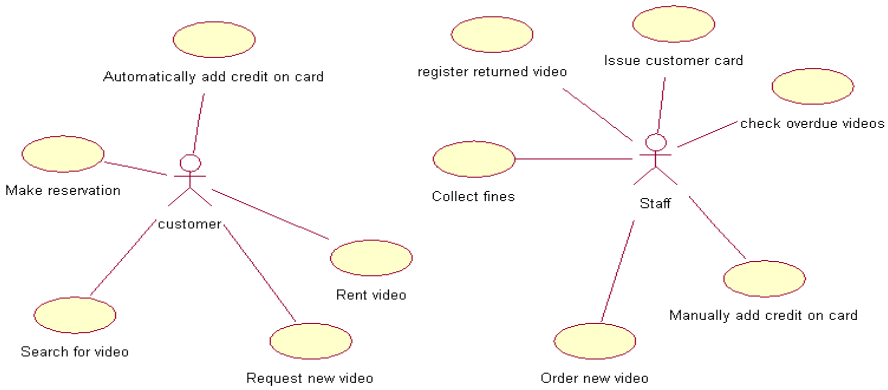

Figure 2. The Use Case Diagram created by the participants in Test 1.

The tests presented have been designed to follow the six phases in the RECOCASE methodology. Tests 1, 2 and 3 were conducted in three sessions. Test 4 consisted of only one 1.5 hour session where the participants received the same introduction as in session 1 in tests 1 and 2 without the description regarding the RECOCASE tool. They also did not follow the RECOCASE methodology. In the studies the tool has not been separated from the methodology, but is used to facilitate

the process. In Test 1, 2 and 4, the participants were presented the AVS problem (figure 1) and brainstormed to identify use cases and created a shared use case description on the board. Before the second session in Test 1, 2 and 3, the GF entered the project data and registered users using the RECOCASE tool. The second session was an individual session where the test participants entered their use case descriptions into the tool and completed a survey rating the tool. The test participants then had a break while the GF created line diagrams by combining the different use case descriptions. In session 3 the group and the GF met and discussed the diagrams. A final shared use case description was created by reconciling the use case sentences.

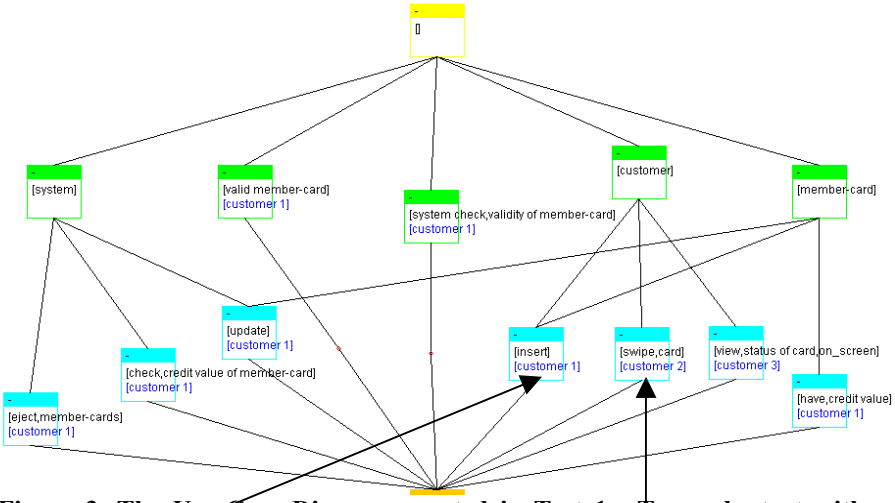

Figure 3. The Use Case Diagram created in Test 1. To read, start with a bottom node to find the owner of the sentence. Pick up parts of the sentence by following paths up. Apart from other differences/ similarities we can see that customer1 thinks the card should be inserted, customer 2 sees that the card should be swiped. They may decide to not commit to the input device at this stage (in keeping with customer 3's view but not shown on this diagram) or to decide the nature of the interaction now.

4. Results of the four evaluation studies

Observations of each session by the authors, remarks of the GF, and analysis of resulting use case descriptions have been used to complement data from questionnaires. The data collected concerned the process and the tool, however, as the tool was used as part of the process the results for the process are affected by the tool. We collected much more data than we are able to present in this paper. Due to space limitations, only partial results for Test 1 that were mostly focused on the process will be explained with figures and screenshots. As Test 4 had a different format some description of that test is also given.

4.1. Test 1

In session one the participants developed the use case diagram in figure 2. In session 2 each individual asynchronously entered use case descriptions into the RECOCASE tool. The participants were asked to write the description for the 'rent video' and 'issue customer card' use cases (figure 2). This was their first trial with the RECOCASE tool. They had received a 10 minute presentation of the tool in session 1, but this was a bit short and the participants had some problems in the beginning to follow the logical path in the application and to locate the command buttons. However, the test participants learned quickly how to use the application, and looked much more confident when inserting the second use case description. Tables 1-4 show the individual and shared descriptions from Test 1 for the "rent video" use case.

Table 1. The use case description for "Rent Video" written by Participant 1 in Test 1.

Step	Action
1	Customer selects the video
2	Customer scans the video barcode
3	Customer swipes the card
3.a.	If the card has enough credits
3.a.1.	The credit in the card is deducted
4	Customer receives the receipt
5	Customer receives the rent duration statement

Table 2. The use case description for "Rent Video" written by Participant 2 in Test 1.

Step	Action
1	The customer insert the member card into the machine
1.1	The system read the data in the card
2	The customer search for video titles
2.1.	The system display the current availability of the video to the customer
3	The customer picks the video the customer wants
4	The system reserved the video for the current customer
5	The customer checks out

Table 3. The use case description for "Rent Video" written by Participant 3 in Test 1.

Step	Action
1	The customer presents the rental card to the interface device of the system
2	The customer presents the tape to the interface device of the system
3	The system attach identification information of the tapes to the customers record
4	The system calculates the due date
5	The system debits the credit of the customer account
6	The system prints a tax invoice

Table 4. The final shared description for the "Rent Video" use case in Test 1.

Step	Action
1	Customer swipes the card
2	The system read the data in the card
3	The system displays the credit value
4	Customer scans the video barcode
5	The system shows the amount payable
6	If the card has enough credits
7	The system calculate the due date
8	The system attach identification information of the tapes to the customers record
9	The system debits the credit of the customer account
10	The system displays the current balance
11	Customer receives the rent duration statement
12	The system prints a tax invoice

Before the group met again in session 3, the GF looked at the different use case descriptions, identify and register synonyms, and create line diagrams that combined the different sentences. There were too many sentences to create only one diagram, and the GF created two diagrams, one for sentences describing system interaction (sentences including the word 'system') and one for the actor; 'customer'. Some sentences regarding the use of the customer card did not contain the words 'system' and 'customer', so a diagram with these sentences was also constructed.

In the third session the test participants gathered around the screen to look at the diagrams and decide what sentences to use to create a final shared use case description. If the participants wanted to remove sentences from the diagrams, they were marked 'Ignore', and if the participants could not agree on what action to take, the sentence was marked 'Delay'. The participants decided to not use substeps in the final use case description and this is also in compliance with style 1 in the CP Use Case Writing Rules by Cox et. al [1]. Figure 3 shows a final diagram from Test 1.

Phase 6 in the RECOCASE process model involves evaluation of the final shared use case. The tool can calculate if the viewpoints have become more similar by determining node by node and agent by agent how many attributes are shared by different agents divided by the total number of words that belong to that sentence. Contrasting sentences score 0%, and sentences in complete consensus score 100%. The scores for each sentence for each agent are averaged to produce an overall measure of similarity. The percentage of concepts shared between the different viewpoints and the final use case description can also be shown (see Figure 4).

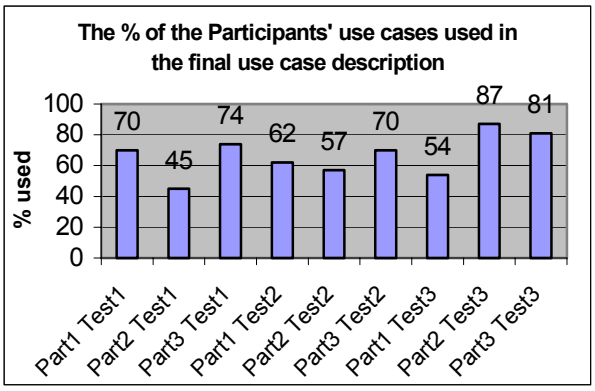

Figure 4. The percentage of individual descriptions retained in the shared use case.

4.2. Test 4

In test 4 the participants were asked to draw a use case diagram for the test case problem in Figure 1 and to write a use case description together for the use case "rent video". The group did not use the RECOCASE tool, and the use case diagram and the use case sentences were written on a board. All the participants said they agreed or strongly agreed to that they were confident in using computers. Participant 1 and 2 wrote they were confident in writing use cases, and participant 3 ticked 'undecided' in response to this statement. The participants attended one session which lasted for 75 minutes and included a short presentation of the RECOCASE methodology and the motivation behind our approach. An example problem with a possible solution was also shown to the participants before they were presented with the test case problem. As in Tests 1, 2 and 3, the participants thought about the problem before the drawing began. The GF drew the diagram and wrote the use case sentences according to the

participants' directions. Participants 1 and 2 started explaining what they wanted in the use case diagram, while participant 3 did not say anything. When the GF asked him if he had any suggestions/comments he said he disagreed with what the others had said and told how he wanted the diagram to look. His diagram did not follow traditional drawing standards, and it did not illustrate the use case problem he had been presented. When the GF explained that the group would have to solve the problem given and also follow drawing standards, the participant reacted by distancing himself from the group. He was only spoke when the GF addressed him. It seemed that he did not care, and in the post test survey he ticked 'agree' to all the statements. The two other participants were both active during the drawing of the use case diagram and during the writing of the use case description.

5. Study findings

5.1. Analysis of process outputs: The use case descriptions

Table 5 shows the participants' length of use case 1 compared to other participants and the final shared use case description. In tests 1 and 2, all the participants wrote shorter and less detailed use case descriptions than the final shared use case description. This is in contrast to Test 3, where two of the test participants actually wrote longer use case descriptions than the final shared use case. When the use case sentences in test 3 were reconciled, it was discovered that parts of the long use cases were too detailed and actually described a low level design. Some of the sentences could also be classified as belonging to other use cases, and were removed from the diagrams by the participants. However, the participants were more confident in writing use case descriptions and also more effective at using the tool.

Table 5. The number of sentences in the participants' use case descriptions and the final shared use cases.

	Number of Sentences in Use Case Description			
Test	**Part 1**	**Part 2**	**Part 3**	**Shared Use Case**
Test 1	7	7	6	12
Test 2	11	7	6	15
Test3	17	6	20	16
Test 4	NA	NA	NA	10

Test 4 was conducted with the same test problem as in Tests 1 and 2 but without using the RECOCASE tool. The resulting use case description from Test 4 consists of 10 lines and is shorter than any of the other use case descriptions. A comparison of the different use case descriptions showed that the description produced in Test 4 described the same main parts of the use case as Test 1 and 2, but provided fewer steps and less detail. These results support our expectation that RECOCASE provides a means to gain a longer and more detailed use case description. A possible reason for this can be that by using the tool individually the participants think on their own before they meet the group to decide the final description and that this prevents group thinking.

The most notable difference between writing use case descriptions with the tool (tests 1, 2 and 3) and without the tool (test 4) concerned the group process. Using the tool, participants could write whatever they wanted at their own pace and using their own terms. Their sentences would be displayed in diagrams together with the other participants' sentences. This may have given them a sense of ownership of their

sentences and resulted in increased involvement in the discussion (and hopefully the project overall) compared to group brainstorming of use cases as in test 4. We would need to conduct further studies without the tool to see if similar behaviour emerged. Our focus in these studies had been on the tool and the RECOCASE process. The final shared use case descriptions are created by reviewing all of the individual sentences combined in a diagram. Figure 4 shows that a major part of each individual use case description is retained in the final description.

5.2. Analysis of the RECOCASE Tool

The test participants in Tests 1, 2 and 3 were asked to rate their agreement to statements regarding the usability of the RECOCASE tool. The participants could tick 'no opinion', 'strongly disagree', 'disagree', 'undecided', 'agree' and 'strongly agree', and these ratings are assigned to the values 0-5. The results shows that the participants liked the application and found it easy to learn. However, the efficiency ratings were quite low during the first test. Several modifications to improve the efficiency were made to the tool based on the feedback from the test participants. Some command buttons were moved and pop up windows had been removed to make the process of entering use case sentences quicker. The ratings improved in Test 2, but more feedback led to added changes and the ratings increased again in Test 3. Participants generally found it easy to recover from errors, except two participants who unfortunately experienced premature termination of the application while entering their viewpoints. Since the test participants in Test 3 had used the tool previously in Test 1, they were asked to rate memorability. All agreed or strongly agreed that it was easy to remember how to use the tool.

5.3. Analysis of the RECOCASE Methodology

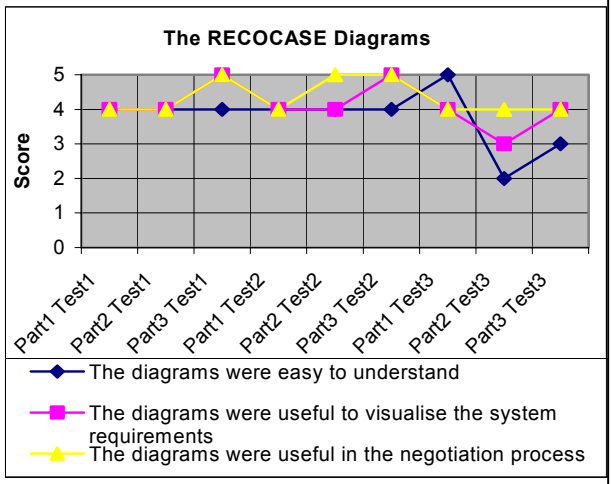

Figure 5. The test participants' ratings of the diagrams created by the RECOCASE tool. The diagrams are used in the reconciliation process in session 3.

The possibility of visualizing use case sentences by using line diagrams is an important part of the RECOCASE tool. The test participants used the line diagrams as a starting point for the discussion of the use case description. The GF guides the group members in the reconciliation process, but it is essential that all the participants understand how to read the diagrams. As shown in Figure 5, in the survey all the participants in Test 1

and 2 answered that the diagrams were easy to understand. Surprisingly, they found the diagrams harder in Test 3. The reason for this is likely the increased number of sentences the group members had to reconcile. Table 5 shows that the participants wrote longer use case descriptions and this again made it necessary to construct more diagrams. Four diagrams were constructed compared to three in Test 1 and two in Test 2. The diagrams in Test 3 were larger than those in the other tests and this probably decreased understandability. The participants in Tests 1 and 2 found the diagrams useful in visualizing the system requirements, and they also thought they were useful in the negotiation process. Again the ratings are lower in Test 3, confirming that smaller diagrams improves the diagrams' usefulness.

Some of the participants found it difficult initially to write use case descriptions but participants were usually faster and more confident when writing the second description. The group members in Test 1 and 2 found it easy to reconcile the use case sentences by using the RECOCASE methodology and tool. Again, the participants in Test 3 described the process as more difficult, probably because of the size and the number of diagrams.

The second group of participants strongly agreed that the capture of multiple viewpoints was very useful in understanding the complete set of requirements. The first group agreed and participant one strongly agreed after they had performed the second study. It is interesting that in all but one response (out of the nine) the use of multiple viewpoints was seen to be at least as useful as the use case approach itself. From our observation of participants during the process it was clear that working on your own use case and then needing to defend it greatly improved participation and resulted in all participants (to varying degrees of course) becoming engaged in the development of the set of requirements.

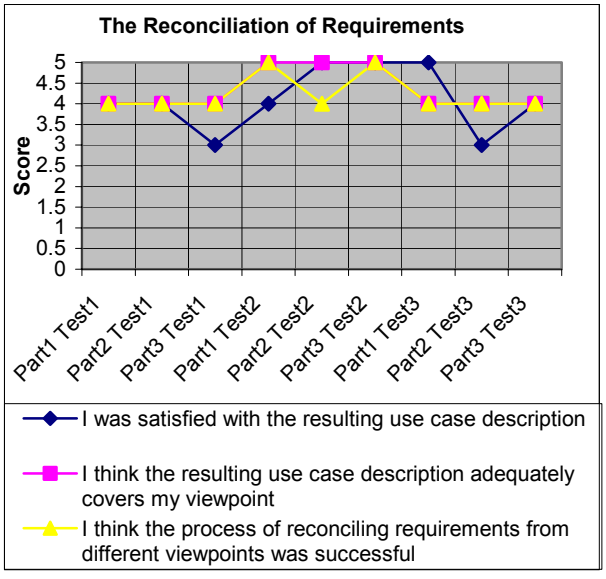

Figure 6. The participants' ratings of the final shared use case and the reconciliation process

Even though all participants in Tests 1, 2 and 3 expressed that the resulting use case description covered their viewpoint, not all were completely satisfied with the resulting use case description (figure 6). Perhaps while they agreed that the description covers their viewpoint, they may have felt their own description was better. All agreed or strongly agreed to the statement saying the process of reconciling requirements from different viewpoints had been successful. This is a very positive and encouraging result.

6. Discussion and Conclusion

From a number of previous studies involving a total of around 450 second and third year computing students over the past two years we have found that students prefer to use the line diagram for comparing requirements and that they were able to do so more quickly and accurately than with comparisons of textual descriptions. With such promising results we wanted to test whether the group process we had developed was practical, would produce a shared set of requirements and whether that set was more representative and comprehensive than would have been achieved through the normal process of defining use case descriptions.

As presented, our results show that each final and shared use case description was longer than the individual use case descriptions (once invalid sentences were removed). All participants were satisfied with the process and the outcome and were able to create a shared use case description using the technique. The most striking difference was found in the attitudes and cohesion of the groups that used the process compared to the group that did not. While we only performed one study that did not use RECOCASE, as our goal was primarily to look at our methodology and tool, we found that it was impossible with that group to develop a use case description that represented the whole group's view. Given that the system being specified is intended to represent the needs of all stakeholders the impact of such group dynamics is bound to result in a specification that is not truly representative. The ill-will that can be generated from such a process may also possibly result in lack of acceptance of the system or even open hostility and sabotage of the delivered system even if it meets the requirements.

The next step in this project is to perform evaluations with RE practitioners in industry and to make the appropriate refinements based on our findings.

References:

[1] Cox, K., Phalp, K., & Sheppard, M. (2001) Comparing Use Case Writing Guidelines, *7th International Workshop on Requirements Engineering: Foundation for Software Quality (REFSQ'01) , Interlaken, Switzerland,* June 2001, pp. 101-112.

[2] Easterbrook, S. *Elicitation of Requirements from Multiple Perspectives PhD Thesis*, Dept. of Computing, Imperial College of Science, Tech.and Med., Uni. of London, London.

[3] Ramesh, B. & Dhar, V. (1992) Supporting Systems Development by Capturing Deliberations During Requirements Engineering, *IEEE Transactions on Software Engineering*, 18(6), pp. 498-510

[4] Richards, D., (2003) Merging Individual Conceptual Models of Requirements, Special issue on Model-Based RE for the International Journal of Requirements Engineering.

[5] Richards*, D*. and Boettger, K (2002) Representing Requirements in Natural Language as Concept Lattices, *Proc. 22nd Annual Int. Conf. of the British Computer Society's Specialist Group on Artificial Intelligence* (SGES), (ES2002), Cambridge, UK, Dec 10-12, 2002.

[6] Richards, D., Boettger, K and Aguilera, O. (2002) A Controlled Language to Assist Conversion of Use Case Descriptions, *15th Australian Joint Conference on Artificial Intelligence* Canberra, Australia, December 2nd - 6th, 2002

[7] Wille, R. (1992) Concept Lattices and Conceptual Knowledge Systems *Computers Math. Applic. (23)* 6-9: 493-515.