Requirements Analysis through Viewpoints Oriented Requirements Model (VORD)

Ahmed M. Salem
Computer Science Department
California State University, Sacramento
Sacramento, CA 95819 USA
Email: salema@ecs.csus.edu

***Abstract*— This paper describes an extension to the Viewpoints Oriented Requirements Definition (VORD) model and attempts to resolve its lack of direct support for viewpoint interaction. Supporting the viewpoint interaction provides a useful tool for analyzing requirements changes and automating systems. It can also be used to indicate when multiple requirements are specified as a single requirement. The extension is demonstrated with the bank auto-teller system that was part of the original VORD proposal.**

Keywords: Software Requirements; Requirements Modeling; Functional Requirements

I. INTRODUCTION

The Viewpoints Oriented Requirements Definition (VORD) was proposed by [1] by Kotonya and Somerville as a method to tackle requirements engineering from a viewpoint level. A significant part in the software development process today, is not anymore programming, designing or testing, but requirement analysis. Interactive systems, whose operations involve a degree of user interaction, have a serious problem in identifying all the clients' needs. At the same time the analyst has to be sure that all needs are recognized in a valid way. The VORD method is useful in detecting these user needs, and also identifying the services that a user expects from the system [10]. It provides a structured method for collecting, documenting, analyzing, and specifying viewpoints and their requirements. Viewpoints map to classes of end-users of a system or to other systems interfaced to it. The viewpoints that make up the core model are known as direct viewpoints. To allow organizational requirements and concerns to be taken into account, viewpoints concerned with the system's influence on the organization are also considered. These are known as indirect viewpoints [2]. When describing the VORD model, its creators identified a limitation that it does not explicitly support the analysis of interaction across and within all of the viewpoints. This paper will expand upon the Viewpoints Oriented Requirements Definition (VORD) process by modifying the approach; so that it supports viewpoint interaction without any degradation to the existing framework. In addition to proposing a theoretical solution to support viewpoint interaction, we will show how the solution works by using practical examples. Much of the breadth of this paper consists of working out the practical application of our model extension.

II. BACKGROUND

In a follow-up piece, [2] on the VORD model, one of the original authors noted that the viewpoint interaction limitation still existed. After researching other papers on viewpoint interaction, we were able to find one that closely expanded on this topic, however the VORD model was not updated directly. The paper [3] was on the VISION model proposed by Araújo and Coutinho. Their approach took ideas from the VORD and PREVIEW [4] models, then incorporated viewpoint associations, UML modeling, and aspectual use cases. They have identified some basic methods of identifying and documenting viewpoint relationships that we would like to incorporate into our enhancement. We will expand the viewpoint interaction portion of their solution, and then use it to enhance the VORD method.

Other authors have also used the VORD process model during requirement analysis; a paper by author Zeljka Pozgaj, clearly explains the three steps of the VORD using a Stakeholder example. The UML diagrams illustrate the viewpoint and service interactions [10]. There have been other methods of incorporating multiple viewpoints into requirements engineering other than the VORD model. Lee's Proxy Viewpoints Model-based Requirements Discovery (PVRD) [5] methodology is especially designed for working from "legacy" SRS documents. Others use viewpoints as a tool for addressing possible inconsistencies in requirement specifications such as Greenspan, Mylopoulos, and Borgida [6] and Nuseibeh and Easterbrook [7].

III. VORD PROCESS MODEL

To gain an appropriate understanding of the process extension, it is necessary to provide a brief description of the VORD process model itself. The VORD process model is designed to elicit, analyze, and document the requirements for a Service-Oriented System (SOS). It specifically attempts to look at all the entities that will interact or otherwise use the services of the system. The requirement sources may be from stakeholders, other systems that interface with the proposed system, or other entities in the environment of the proposed

system that may be affected by its operation. Each requirement source is then considered to be a *viewpoint*.

VORD defines two classes of viewpoints:

1) Direct viewpoints: These correspond directly to clients in that they receive services from the system and send control information and data to the system. Direct viewpoints are either system operators/users or other sub-systems, which are interfaced to the system being analyzed.

2) Indirect viewpoints: Indirect viewpoints have an "interest" in some or all of the services which are delivered by the system but do not interact directly with it. Indirect viewpoints may generate requirements which constrain the services delivered to direct viewpoints [9]. Each viewpoint has a relationship with the proposed system based upon its needs and interactions with the system. The model assumes that if all the viewpoints have been analyzed and specified, then all the system's requirements would also have been analyzed and specified.

The VORD process model is shown in Fig. 1. The first three iterative steps are:

1) *Viewpoint identification and structuring*
2) *Viewpoint documentation*
3) *Viewpoint requirements analysis and specification*

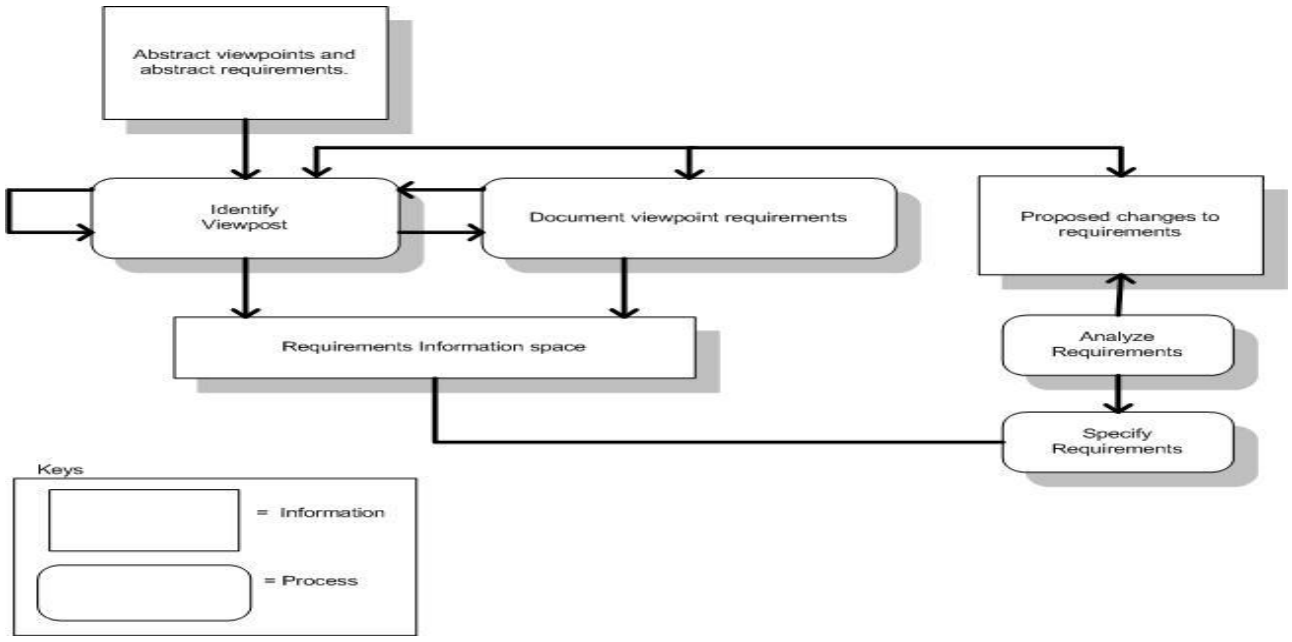

Figure 1. The VORD Process Model

The first step, viewpoint identification and structuring, is concerned with identifying relevant viewpoints in the problem domain and structuring them. The second step is concerned with documenting the viewpoints identified in step 1. Viewpoint documentation consists of documenting the viewpoint name, requirements, constraints on its requirements and its source. Viewpoint requirements include a set of required services, control requirements and set of non-functional requirements. The last step is concerned with analyzing, and specifying the functional and non-functional viewpoint requirements in an appropriate form [9].

IV. Proposed VORD Process Model Extension

Our extension is an iterative process that takes place after step three. Our extension has three steps:

1) *Requirement to viewpoint mapping*
2) *Viewpoint interaction analysis*
3) *Viewpoint interaction documentation (interaction matrix)*

The first step of our model extension is to map each requirement to its associated viewpoints. This is actually backward from the VORD model listing viewpoints first and requirements second. This is needed since we assume that the most reliable method of identifying interactions is at the level of required services, control requirements, or set of non-functional requirements. This step is done by first listing all the labeled requirements, both functional and non-functional. Then the associated viewpoints for each requirement are listed.

The second step of our model extension is to determine if any viewpoint interaction exists for each requirement. This is done by analyzing the list created in step one, along with the specification for each requirement. If a requirement has only one associated viewpoint, then we can assume that no

viewpoint interaction takes place (for that requirement), and no further analysis is needed. If there are two or more viewpoints listed, then further analysis is needed. This analysis consists of analyzing the requirement specification and determining if the first viewpoint listed interacts with the second viewpoint listed. This process is continued until all the viewpoints for the requirement have been compared against all the other viewpoints for the requirement. If any interaction is discovered, then the interaction type must be determined. If there is a transitive relationship, i.e. the viewpoints interact but only through another viewpoint, then it is considered to be indirect interaction. If the viewpoints interact directly, then it is considered to be direct interaction. If there is no interaction, then this may be an indication of a *compound requirement*. In this case, the requirement should probably be split up into two or more requirements.

The third step of our model extension is to document the viewpoint interactions discovered in step two. The results are displayed in an interaction matrix that has rows and columns for each viewpoint and the requirement name(s) listed in the corresponding "box". Note that there may be more than one requirement listed in a box since two viewpoints may have interactions in more than one requirement. There may be boxes with no requirements listed since two viewpoints may not always have an interaction.

V. ACTICAL APPLICATION AND ANALYSIS

We devised several examples of viewpoint interactions, and then applied our model extension to them. The reason for this is two fold. The first reason is to provide a means of explaining our proposal. Using examples provide a clear understanding of the theoretical model. The second reason is to actually "test" the practicality of our proposal. Using examples, not only help to show the proposal's strengths and limitations, but also its ease of use.

By continuing the ATM machine case study, we feel that our paper compliments the original proposal; similarly, our theoretical model compliments the original model. The requirements chosen to extend the ATM case study to demonstrate viewpoint interactions are listed in Table I. These requirements are specified to a much less thorough extent than they would be if they were actually part of an SRS document. However, the purpose here is to provide just enough narrative to convey the essence of the requirement.

A. *Example 1: Banking*

The original VORD proposal listed the following viewpoints in the case study.

- Bank manager
- ATM operator
- Home customer
- Customer database
- Foreign customer
- Security officer
- System developer
- Bank policy

For the purposes of clarity, we modified the viewpoint list.

The home customer and foreign customer viewpoints are combined into a single "bank customer" viewpoint. The reason is that no distinction is necessary in the requirement examples that we have chosen. We also added the Bank employee viewpoint since the ATM operator viewpoint is only concerned with stocking the ATM with cash and starting and stopping its operation. The result of applying step one of our model extensions is listed in Table II.

Step two requires more detailed analysis. Within each requirement, each viewpoint is analyzed and compared to the other viewpoints. Listed below is the viewpoint interaction analysis for each requirement. The general approach is to compare each viewpoint to all the other viewpoints in the requirement. Then, determine if any interaction takes place. If interaction takes place directly, then it is considered a direct interaction. If two viewpoints interact, but only through other viewpoints, then it is considered an indirect interaction.

TABLE I. EXTENDED CASE STUDY REQUIREMENTS

Requirement Name	Description
R1	Bank manager approves the bank customer to withdraw funds above the daily withdraw limit.
R2	Bank employee reset's bank customer's forgotten PIN number.
R3	Bank employee provides replacement card to a bank customer. (Card is lost, stolen, damaged etc.)
R4	Bank customer notifies bank employees of ATM machine problems. (Out of money, malfunctioning etc.)
R5	Bank employee issues a ATM card to a new customer.
R6	Bank customer reports unauthorized withdraw from ATM to bank employee.
R7	Bank employee notifies bank customer of a newly added ATM machine.
R8	Bank customer makes a deposit.

1) *R1 Viewpoint Interaction Analysis*

The case study of the original VORD proposal listed bank policy as the viewpoint handling for a variety of business rules. Although, not explicitly stated in the original proposal, we are

assuming that a daily ATM withdraw limit per bank customer is part of bank policy.

The bank customer's withdraw is limited by the bank policy. The bank customer requests a bank employee, to permit him to withdraw funds beyond the daily limit. The bank employee then informs the bank manager of the request, along with any relevant justification. The bank manager then modifies the bank policy (granting additional amount to the customer's daily withdraw limit), which allows the bank customer to withdraw additional funds. The bank customer directly interacts with the bank employee with the request to withdraw additional funds.

The bank employee and the bank manager directly interact with the bank customer's request. If the bank manager grants the customer request, then the bank manager interacts directly with the bank policy. The bank customer indirectly interacts with the bank manager through the bank employee.

The bank employee indirectly interacts with the bank policy through the bank manager.

2) R2 Viewpoint Interaction Analysis

The bank customer contacts a bank employee to have a PIN number reset. The bank employee then resets the PIN number. Therefore, the bank employee and bank customer directly interact.

3) R3 Viewpoint Interaction Analysis

The bank customer contacts a bank employee to request a new ATM card. The bank employee then provides the bank customer with a new card. Therefore, the bank employee and bank customer directly interact.

4) R4 Viewpoint Interaction Analysis

The bank customer notifies the bank employee of an ATM machine problem. The bank employee in turn notifies the ATM operator about the problem. After the problem is resolved, the ATM operator notifies the bank employee, who then informs the bank customer who reported the problem. Direct interaction takes place between the bank customer and bank employee (notification). Direct interaction takes place between the bank employee and ATM operator (notification). The bank customer initiates the action that results in the ATM operator performing a function. However, this is done though the bank employee viewpoint. Therefore, the bank customer and the ATM operator indirectly interact.

5) R5 Viewpoint Interaction Analysis

The bank customer opens an account and the bank employee provides an ATM card. Therefore, the bank employee and bank customer directly interact.

6) R6 Viewpoint Interaction Analysis

The bank employee and bank customer directly interact with the notification and collection of facts. The bank employee and the bank manager directly interact with data exchange. The bank manager and bank policy directly interact to determine if the account should be credited for the unauthorized withdraw. The bank policy and the bank customer directly interact with a notification of the manager's decision. The bank customer initiates action that results in the bank manager performing a function. However, this is done through the bank employee viewpoint. Therefore, the bank manager and the bank customer indirectly interact. The bank employee and the bank policy interact indirectly through the bank manager.

7) R7 Viewpoint Interaction Analysis

The bank employee notifies that bank customer of a newly added ATM machine. Therefore, the bank employee and bank customer directly interact.

8) R8 Viewpoint Interaction Analysis

The bank customer is the only viewpoint listed, so it does not interact with any other viewpoints. This requirement was put in place to show that not all requirements will contain viewpoint interaction.

The third step in our model extension is to document the analysis performed in step two. This is displayed in a matrix that consists of columns and rows for each viewpoint. The first column and first row are considered as the "headers" of the matrix as they list the viewpoints used in the example. The intersection of each row and column is a box that represents the interaction between the corresponding viewpoints depicted in the headers. Since each viewpoint is listed twice (first column and first row), there will be two corresponding "boxes" for

TABLE II. REQUIREMENTS TO VIEWPOINT MAPPING

Requirement Name	Description
R1	Bank customer, bank employee, bank manager, bank policy.
R2	Bank employee, bank customer.
R3	Bank employee, bank customer.
R4	Bank customer, bank employee, ATM operator.
R5	Bank employee, bank customer.
R6	Bank customer, bank employee, bank manager, bank policy.
R7	Bank employee, bank customer.
R8	Bank customer.

each viewpoint interaction. The corresponding "box" for each viewpoint interaction lists the requirement name (that is the requirement in which the viewpoints interacted), along with an interaction type designation, "D" for direct and "I" for indirect interaction. Note that the matrix design will include two "boxes" that correspond to the interaction of a single viewpoint (i.e.: bank customer to bank customer). These will be left blank since a viewpoint did not interact with itself in any instance.

The viewpoint interaction matrix created from the analysis in step two is displayed in Table III. The matrix was created using the methodology described in the previous paragraph. The first column and first row list all the defined viewpoints. The corresponding "box" for each viewpoint interaction list all the requirements where the viewpoints interact, along with a designation of "D" for direct and "I" for indirect. For the purposes of brevity, only the viewpoints used in R1 – R8 are shown.

B. Example 2: Library System

By considering the Library System (LS) case study, we propose that our paper compliments the original proposal; similarly, our theoretical model compliments the original model. The requirements chosen to extend the LS case study to demonstrate viewpoint interactions are listed in Example 2 – Table IV. These requirements are briefly described than they would be as part of an SRS document. However, the purpose here is to provide just enough narrative to convey the essence of the requirement.

Viewpoints in Case Study:

- Library Member
- Library Staff
- Library Manager
- Library Policy
- Library IT Administrator
- Library Database

The result of applying step one of our model extensions is listed in Example 2 - Table V.

The library system is then analyzed and the interactions between the viewpoints are listed. The interaction matrix in Example 3 - Table III is then derived from the defined viewpoint interactions.

1) R1 Viewpoint Interaction Analysis

Library staff informs the new library members about the services offered in the library. Library staff and member involve in a direct interaction with each other and therefore direct relationship exists for this requirement.

2) R2 Viewpoint Interaction Analysis

Library member requests the library staff for a book available in another library. Library Staff informs the Library Manager about the request, who in turn refers to the library policy. If available, the library manager arranges for the book and ensures that the member receives it through the library staff. Here, the library member and library manager interact through another viewpoint, that is the library staff and therefore they maintain an indirection relationship. Library member and staff maintain a direct relationship because of direct interaction. The Manager maintains a direct interaction with the library policy since he refers to it.

TABLE III. VIEWPOINT INTERACTION MATRIX

	Bank Customer	**Bank Employee**	**Bank Manager**	**Bank Policy**	**ATM Operator**
Bank Customer		R1 D R2 D R3 D R4 D R5 D R6 D R7 D	R1 I R6 I	R1 D R6 D	R4 I
Bank Employee	R1 D R2 D R3 D R4 D R5 D R6 D R7 D		R6 D	R1 I R6 I	R4 D
Bank Manager	R1 I R6 I	R6 D		R1 D R6 D	
Bank Policy	R1 D R6 D	R1 I R6 I	R1 D R6 D		
ATM Operator	R4 I	R4 D			

TABLE IV. EXTENDED CASE STUDY REQUIREMENTS

Requirement Name	Description
R1	Library staff, library member.
R2	Library staff, library member, library manager, library policy.
R3	Library staff, library member, library manager, library policy.
R4	Library staff, library member, library database.
R5	Library staff, library member, library IT administrator.
R6	Library staff, library member, library database.
R7	Library member.
R8	Library member, library policy.

TABLE V. REQUIREMENT TO VIEWPOINT MAPPING

Requirement Name	Description
R1	Library staff informs new library members services offered in the library.
R2	Library members request books from another library.
R3	Library manager approves library members for checking out books more than prescribed limit.
R4	Library members informs non availability of the books to the staff
R5	Library members notifies library staff problem in library website.
R6	Library staff updates member record in the database after member pays back late fee.
R7	Library member checks in the book.
R8	Library member talking in his/her phone in silent place of library.

3) R3 Viewpoint Interaction Analysis

Library member requests the library staff to check out more books than the prescribed limit. Library staff informs the above issue to the manager who in turn refers to the library policy and allows the member to checkout more books if library policy permits it under special circumstances. Here, the library member and manager maintain an indirect relationship whereas the library manager and library policy, library member and library staff, maintain a direct relationship. Since the policy is referred to, to clarify the doubt of the library member, the member and staff indirectly interact with the policy as well.

4) R4 Viewpoint Interaction Analysis

Library member informs the non availability of the book to the staff. Library staff checks in the library database about the availability and informs the member about the availability of the book. The library member and staff thus have a direct interaction. Here, the library member and library database have an indirect relationship because they interact through the library staff.

5) R5 Viewpoint Interaction Analysis

Library member notifies the problem in the library website to the library staff who in turn contacts the Library IT administrator to resolve the issue. Here, the library member and IT administrator interact indirectly through the library staff and thus maintain an indirect relationship.

6) R6 Viewpoint Interaction Analysis

Library staff updates the member record in the database after the member pays back the late fee. Here, library member and database involve in an indirect interaction. The Library member pays the late fee directly to the staff and therefore the relationship is direct.

7) R7 Viewpoint Interaction Analysis

Library member checks in the book. Here, the only viewpoint is the library member, who in this case does not interact with any other view points. This requirement was put in place to show that not all requirements contain viewpoint interaction.

8) R8 Viewpoint Interaction Analysis

Library member talking over his/her phone in the silent study place of the library is the compound requirement because requirement's viewpoints such as member and policy do not interact with each other. In order to resolve this issue, the compound requirement needs to be split into multiple requirements. The above requirement can be split into the following:

Library member talking outside library.	Library member.
Maintaining silence in silent study place.	Library policy.

VI. ANALYSIS

The lack of explicit support for viewpoint interaction is listed in the limitations section of the original proposal as an area for further research. Our extension to VORD provides that direct support for viewpoint interaction. Due to the complex nature of requirements engineering, viewpoint interaction is a common occurrence. Therefore, our proposal strengthens the VORD model's ability to cope with a problem that it previously could not directly address.

By providing direct support for viewpoint interaction, our model extension would be useful for automating legacy systems. For example, a bank may wish to lower operating costs by automating certain customer operations. By analyzing the viewpoint interaction matrix, all interactions to the bank employee viewpoint are clearly defined. The bank may choose to automate certain operations provided by the employee such as the notification to the bank manager in R6. A possible cost effective solution may be used to provide a web-based user interface that allows the bank customer to send the relevant information to the bank manager. This can be observed in terms of the Library System example as well; by analyzing the viewpoint interaction matrix, all interactions between each of the viewpoints are clearly defined. The interaction matrix gives an overall view of the direct and indirect interactions between the viewpoints. By analyzing the interaction matrix, the library may choose to automate certain operations that would aid in reducing the number of interactions yet perform the required task. For instance, the library may choose to automate operations such as the interactions between the library employee and library manager in R2 and R3. This interaction is merely for a notification purpose and hence can be done through other means such as sending an e-mail. By automating such actions, the entire system can be reduced of a number of interactions. Thus overall, we see the library system uses 11 direct interactions and 8 indirect interactions between its viewpoints. By our simple analysis, we observed that automation of certain processes could reduce the number of interactions, making the system run more efficiently and in a less complex manner. The proposed extended VORD process model simplifies the entire process; it not only links the viewpoints but also presents the interaction matrix.

This model would also be useful when analyzing the effect of modifying legacy systems. It could help to determine which viewpoints may be affected if a specific requirement is modified. At the very least, it would list the viewpoints to re-analyze.

Another result of our extension is the ability to expose probable compound requirements. By analyzing the type of interaction between viewpoints (direct/indirect), it can be determined if two viewpoints would not interact at all within a certain requirement. This may be an indication that the specified requirement may actually contain multiple requirements. The remedy would be to specify the requirement further into two or more requirements. This effect is limited to those requirements that have two or more viewpoints. Our extension will not indicate if a compound requirement is specified for requirements that only affect one viewpoint.

VII. LIMITATIONS AND FUTURE WORK

The original VORD model was deliberately restricted to a service-oriented view of systems. Therefore any restrictions of the service-oriented systems (SOS) paradigm will also affect our extension of VORD. The VORD authors did not consider this to be a serious limitation, as they believed that most systems can be regarded as providing services of some kind to their environment.

One limitation of the original VORD model that was not

TABLE VI. VIEWPOINT INTERACTION MATRIX

	Library Member	**Library Staff**	**Library Manager**	**Library Policy**	**Library Database**	**Library IT Administrator**
Library Member		R1 D R2 D R3 D R4 D R5 D R6 D	R2 I R3 I	R2 I R3 I	R4 I	R5 I
Library Staff	R1 D R2 D R3 D R4 D R5 D R6 D		R2 D R3 D	R2 I R3 I	R4 D R6 D	R5 D
Library Manager	R2 I R3 I	R2 D R3 D		R2 D R3 D		
Library Policy	R2 I R3 I	R2 I R3 I	R2 D R3 D			
Library Database	R4 I	R4 D R6 D				
Library IT Administrator	R5 I	R5 D				

addressed with our extension is the control issues associated with concurrency. The VORD model addresses the process of requesting and responding to services as a linear flow. However, it does not address services provided concurrently to separate entities at the same time. Since, we did not address this issue with our model extension, this limitation still exists.

Since the practical examples continue upon the ATM machine case study, we cannot accurately predict at this time how our model extension will work with other types of systems. However, we feel that the proposed extension should work with other types of systems that use the SOS paradigm.

This paper does not address conflict resolution with viewpoint interaction. That is, if any viewpoint interactions resulted in a conflict, that conflict would need to be resolved when specifying the requirements. The original VORD proposal directly addressed conflict analysis; however, it was limited to conflicts within a single viewpoint not directly addressing conflicts across viewpoints. We recommend that further research be performed in this area. A possible place to start is with goal-oriented analysis [8]. Goal-oriented analysis provides a mechanism for finding alternatives in requirements. It also provides a method for "weighing" each alternative to determine which one best fits the customer's goals. This is useful for finding a "middle ground" (skewed toward the customer's needs) when conflicts occur.

VIII. CONCLUSIONS

The VORD model ensures that system requirements rather than high-level system specification or designs are derived [1]. The model is highly regarded in requirements engineering as demonstrated by the frequency that the original proposal is cited. The authors of the VORD model expressed the lack of viewpoint interaction analysis as a limitation of the model. We used this limitation as a starting point for our research. We devised an extension to the VORD model by providing a method to explicitly support viewpoint interaction. We then demonstrated the practicality of this method by applying it to several examples.

REFERENCES

[1] G. Kotonya and I. Sommerville. Requirements Engineering with Viewpoints. Software Engineering Journal 1996.

[2] G. Kotonya. Practical Experience with viewpoint-Orientated Requirements Specification. Requirements Engineering Journal 1997.

[3] J. Araújo and P. Coutinho. Identifying Aspectual Use Cases Using a Viewpoint-Oriented Requirements Method, Early Aspects 2003: Aspect-Oriented Requirements Engineering and Architecture Design. Workshop of the 2nd International Conference on Aspect-Oriented Software Development, Boston, USA, 17th March 2003.

[4] I. Sommerville, P. Sawyer, and S. Viller Viewpoints for Requirements Elicitation: A Practical Approach. IEEE International Conference on Requirements Engineering (ICRE 1998).

[5] P. Lee. Viewpoints Model-based Requirements Engineering. 2002 ACM Symposium on Applied computing (2002).

[6] S. Greenspan, J. Mylopoulos, and A. Borgida. On Formal Requirements Modeling Languages: RML Revisited, Proceedings of the 16th international conference on Software engineering (1994).

[7] B. Nuseibeh and S. Easterbrook. The Process of Inconsistency Management: A Framework for Understanding, Proceedings of the First International Workshop on the Requirements Engineering Process (REP'99), 2-3 September 1999.

[8] J. Mylopoulos et al. Exploring Alternatives During Requirements Analysis, IEEE Software, January/February 2001, pp. 2 – 6.

[9] Gerald Kotonya and Ian Sommerville. Integrating Safety Analysis and Requirements Engineering, Proceeding of ICSC'97/APSEC'97, pp.259-270

[10] Zeljka Pozgaj. Requirement Analysis Using VORD, 22nd International Conference. Information Technology interfaces, June 13-16, 2000, Pula, Croatia. University of Zagreb, Croatia.