THE FUNDAMENTALS OF PRIORITISING REQUIREMENTS

Frank Moisiadis
Department of Computing, Macquarie University, NSW 2109, Australia.

ABSTRACT

In systems development projects the activity of prioritising requirements is carried out during the requirements engineering phase. The current trend is to develop systems that are quick-to-market, bound by strict budget constraints, and released in progressive versions. The inherent risks with this trend include not adequately fulfilling the core requirements or simply missing important ones altogether.

This research paper specifies the fundamental requirements for a prioritisation process. The research considers where prioritisation should take place during the requirements phase, and who should be involved in the prioritisation process. Current techniques such as AHP and QFD are analysed as to how well they satisfy the fundamental needs of a successful prioritisation process.

A framework is described that incorporates the many aspects of prioritising requirements. These aspects include balancing the various viewpoints and goals of the stakeholders, determining the value of each stakeholder's subjective opinion, and aligning the requirements to the business objectives of the systems development project.

The Requirements Prioritisation Tool (RPT), which is based on the framework, is presented and compared with current methods. It is also evaluated with case studies for how well it satisfies the fundamental requirements of an efficient prioritisation process.

INTRODUCTION TO PRIORITISING REQUIREMENTS

Requirements prioritisation is the setting of ranks or ratings of importance to a set of requirements based on certain criteria like goals, risks, quality, use, and according to the viewpoints of various stakeholders.

The notion of releasing progressive versions and updates on products, as well as the rising demand on developers to build systems that go to market much quicker than ever before has led to the need to prioritise requirements at the earliest possible stage in the systems development life cycle.

Many development projects suffer because the most important requirements are not elicited and analysed at the early stages of the system development process. Meeting budgets and deadlines has become more important than designing and implementing an exhaustive yet comprehensive set of requirements. Most customers opt to reduce the number of product requirements that need to be fulfilled in order to meet their business goals.

Prioritising requirements allows customers and developers to discuss and rate candidate requirements. It provides a means to solve conflicts and balance the expectations of the stakeholders for the proposed product. Prioritising requirements provides the necessary information to make negotiations and decisions on which requirements to discard or focus upon. It provides a means to measure agreement and disagreement amongst the stakeholders and help resolve which requirements are important.

Prioritising requirements is an important activity in the requirements engineering lifecycle. It is a recursive activity that is used while the requirement analysts and the other stakeholders (customers, users) discuss and endeavour to reach agreement on which requirements to continue analyzing, modeling, and validating. Even when a set of requirements is agreed upon, prioritisation is still needed when requirements change. The volatility of requirements can be managed with a flexible prioritisation process in the requirements engineering life cycle. [1-6]

FUNDAMENTAL REQUIREMENTS FOR A PRIORITISATION PROCESS

In a prioritisation process, the following requirements should be considered:

- The prioritisation process should avoid the risk of having biased or tainted stakeholder input affect the prioritisation process.
- The prioritisation process should reduce the risk of missing important requirements
- The prioritisation process should provide benefits that should outweigh the resources needed to apply the prioritisation process
- The prioritisation process should have access to information from many different relevant viewpoints.
- The prioritisation process should incorporate non-functional issues and business objectives in assigning priorities to the requirements.
- The prioritisation process should support the evolution of the requirements into appropriate specifications for the system.
- The prioritisation process should produce priorities that can be validated. There should be a rationale behind the assigned priorities for each priority.
- The prioritisation process should be able to handle the volatility of a set of requirements.

The prioritisation process should be able to cope with the varied levels of abstraction in requirements statements.

The Who and When of Prioritising Requirements Prioritising requirements should involve representatives from each group of stakeholders with a vested interest in the success of the development project. Valuable input can be provided by all of the groups. The users can provide valuable information on functionality and user interfaces, whereas the managers can provide input on non-functional issues like security, and performance. The various levels of viewpoints can provide different insights on the same areas of functionality. The managers usually view the system from a much higher level than the users do. The various levels at which each stakeholder perceives and reasons about the system, can provide valuable input into why various functionalities may or may not be important. The scope of information will greatly improve the accuracy of the priorities.

Trying to prioritise requirements from a completed specification document is quite time consuming and difficult for the stakeholders involved. There are usually hundreds and thousands of requirements that need to be rated. The costs and resources would outweigh any benefits the prioritisation process would provide.

Requirements prioritisation should occur during the elicitation and analysis stages of the requirements engineering lifecycle when the number of requirements is still quite small and they are usually expressed in high-level terms. Early on in the requirements engineering process there is usually a small number of requirements that will guide the requirements analyst through to the elicitation of other detailed requirements. Prioritisation of the few high level requirements can be done without too much effort. The set of requirements can be viewed as a whole alleviating the problem of confusing the stakeholders with too many stimuli. Since the requirements at this early stage are usually expressed as high-level product features or functions, the interaction between the various groups of stakeholders can be done quite easily. The prioritisation of the high-level requirements can be done quite frequently as the stakeholders bounce back and forth ideas and try different combinations of requirements on the prioritisation process. The interaction between analysts and customers on the initial requirements using a versatile prioritisation process can prove very helpful in directing the analysis activities along the right track.

Requirements prioritisation should be a progressive co-operative activity with the stakeholders. It shouldn't be unilateral. Each stakeholder will have different priorities. Prioritising requirements should serve as a vital part of helping the stakeholders agree upon which requirements should be analysed and used to design and implement the desired system.

CURRENT APPROACHES – HOW DO THEY SIZE UP?

Some requirements engineers prefer to use Quality Function Deployment (QFD) [7] or the Analytical Hierarchy Process (AHP) [8] to prioritise requirements. Other approaches include the cost-value approach by Karlsson [9], Wiegers' method [10], as well as a multiplicity of ad hoc industrial practices such as team voting [11].

Quality Function Deployment. The QFD process consists of the Voice of the Customers or Customer Attributes, which are driven by the customer goals, the Voice of the Engineer or the Design Parameters, which are the technical measures needed to build the customer requirements, the Relationship Matrix, which describes the relationship between the customer requirements and the design parameters, and the Roof of the House of Quality (HOQ), which is the correlation between the design parameters [12].

Like any process, the quality of the output is only as good as the quality of the inputs and the controls over the process. With QFD it is critical to address the quality of the user requirements gathering methods, and the management of these requirements while building the house of quality. The QFD process consist of the following stages:

- QFD Planning,
- Requirements Gathering,
- QFD Analysis (building HOQ) [13].

The size of the House of Quality is a crucial issue in QFD development projects. The suggested limit is to work with a 30 by 30 matrix. That is only 30 requirements, which makes QFD difficult to manage with large sets of requirements.

Other concerns include:

- The lack of measures to check the honesty of the subjective stakeholders inputs
- The amount of tedious work needed to get the prioritisation results
- The difficulty of changing the user requirements because of the amount of rework necessary.

QFD relies on the user requirements being stable. The volatility of requirements is not really catered for.

Another concern is the rapid transition from requirements analysis to design and technical issues. This rapid transition in QFD from user requirements to technical solutions does not allow for modeling and validation activities to be performed to the requirements. The jump to design issues can complicate the requirements analysis activities, and cause important requirements to be missed. The user requirements are not final and should not be treated as such. Requirements prioritisation should be trying to determine "what" the system should do, without referring to "how" it will do it [3].

There is no straightforward way of handling temporal relationships or dependencies among requirements in QFD. The user requirements are treated as atomic requirements, which is not realistic. Requirements are dependent on each other, and there are inter-relationships that need to be identified and used in the prioritisation process to produce accurate results.

QFD works with ordinal scales of 1-5 and 1-9. There is substantial evidence to prove that ordinal scales are weak in accurately eliciting stakeholder preferences. This important issue of using scales is handled in section 4.1.

Analytical Hierarchical Process AHP relies on the pair-wise comparisons of a set of requirements using a scale of 1-9. The steps of AHP include:

- Structuring a hierarchy from the top (the objectives from a managerial viewpoint) through the intermediate levels (criteria on which subsequent levels depend) to the lowest levels (which usually is a list of the alternatives).

- Construct a set of pair-wise comparison matrices for each of the lower levels. One matrix for each element in the level immediately above. A element in the higher level is said to be a governing element for those in the lower level. In fact, every element in the lower level affects every element in the upper level. The elements in the lower level are then compared to each other based on their effect on the governing element above. This yields a square matrix of judgments. The pair-wise comparisons are done on a scale of 1 to 9 showing which requirement is more important than the other and to what degree. (1=equal importance, 3=slighter more important, 5=essentially more important, 7=demonstrated importance, 9=extremely more important). If requirement A is slighter more important than B, then 3 is entered in row A, column B and the reciprocal is assigned to row B, column A.
- There are n(n - 1)/2 comparisons required to be made for each matrix, where n is the number of requirements used in the .matrix.
- The consistency is determined using the eigenvalue [8, 14].

Expert Choice™ is a group meta-decision support software product based on the decision-making methodology AHP. The Expert Choice method works by guiding decision-makers through a series of pair-wise comparisons to derive priorities for decision objectives and options. Expert Choice lets you combine and synthesize the judgments of any subset of decision-makers or the entire group to provide the full spectrum of different perceptions of a problem.

AHP suffers from the problems caused by trying to model requirements into a dominance hierarchy or network. This can be quite a time-consuming and tedious exercise. AHP can only handle inter-relationships of requirements under a higher element of the hierarchy. Similar to QFD, dependencies amongst requirements are not really handled as requirements can appear under many threads or concepts, and can't realistically be restricted to such modelling techniques.

Another concern is the restriction Saaty imposes on how many requirements should be used in a matrix. Saaty states that large subsets of requirements (anything over seven requirements to a set) results in computational explosions and is more prone to produce inconsistencies in the results [8].

There are no measures in place to check the motivation behind the stakeholders' preferences, and changes in requirements result in (n-1) pair-wise comparisons that need to be redone for each new requirement.

There aren't any non-functional or business requirements used to help the stakeholders choose their preference, and there is no rationale provided for the preferences or results. Finally the use of a 1-9 scale causes complications for many stakeholders as stated in the case studies performed in [15] and by evidence presented in section 4.1. The use of a nine-point scale including fractions has not been empirically proved to be optimal in system development projects.

Karlsson's Cost-Value Approach

Karlsson has adopted parts of the AHP concept and developed a cost-value approach for prioritizing requirements. He compared the requirements in a pairwise manner with regard to their relative value of importance and relative cost to implement. The relative intensities that resulted were plotted on a cost-value diagram that revealed which requirements were better to use. He then improved the technique by reducing the number of comparisons needed by implementing local and global stopping rules [5].

The cost-value approach suffers from the same weaknesses as AHP. Psychologists [16-19] state that doing pairwise comparisons gives opportunity for deviation from strictly consistent rank order, since not all the stimuli are present for simultaneous observation and comparison. The use of a 1-9 scale, and the lack of handling requirements dependencies and input from stakeholders with hidden agendas weakens the effectiveness of the approach. The use of business and non-functional issues like cost and value is encouraging. However, using more than a few non-functional issues overburdens the effort required to get the stakeholders to rate the requirements based on each one of these non-functional issues.

Wiegers Prioritisation Matrix Wiegers [10] describes a semi-quantitative analytical approach to requirements prioritisation. The approach distributes a set of estimated priorities across a continuum, rather than grouping them into just a few priority levels. This prioritisation scheme borrows from the QFD concept of customer value depending on both the customer benefit provided if a specific product feature is present and the penalty paid if that feature is absent. A feature's attractiveness is directly proportional to the value it provides and inversely proportional to its cost and the technical risk associated with implementing it. All other things being equal, those features that have the highest risk-adjusted value/cost ratio should have the highest priority.

The approach suffers from similar issues that AHP and QFD do not adequately handle. The notion of checking the subjective input, handling dependencies, and catering for requirements volatility is lacking. The stakeholders are restricted to using an ordinal 1-9 scale, and the approach only incorporates a limited number of non-functional and business issues in its prioritisation technique.

Feature	*Relative Benefit*	*Relative Penalty*	*Total Value*	*Value %*	*Relative Cost*	*Cost %*	*Relative Risk*	*Risk %*	*Priority*
1. Query status of a vendor order	5	3	13	8.4	2	4.8	1	3.0	1.345
2. Generate a Chemical Stockroom inventory report	9	7	25	16.2	5	11.9	3	9.1	0.987
3. See history of a specific chemical container	5	5	15	9.7	3	7.1	2	6.1	0.957

Figure 1. Wiegers approach to prioritising a set of features for a Chemical Tracking System.

FRAMEWORK FOR PRIORITISING REQUIREMENTS

The requirements prioritisation framework and tool proposed in this paper endeavour to assist in solving the apparent weaknesses in current approaches to prioritising requirements.

Prioritising requirements involves complex multi-faceted issues. The issues handled in the requirements prioritisation framework include:

- Eliciting the stakeholders' business goals and non-functional concerns for the system development project and producing a ranked set of business goals
- Rating the stakeholders involved in the project by building stakeholder profile models that can be used to assess the subjective input of the stakeholders upon the business goals and requirements
- Allowing the stakeholders to rate the importance of the requirements, the

business goals, and the influence factor between the business goals and

the requirements using the fuzzy

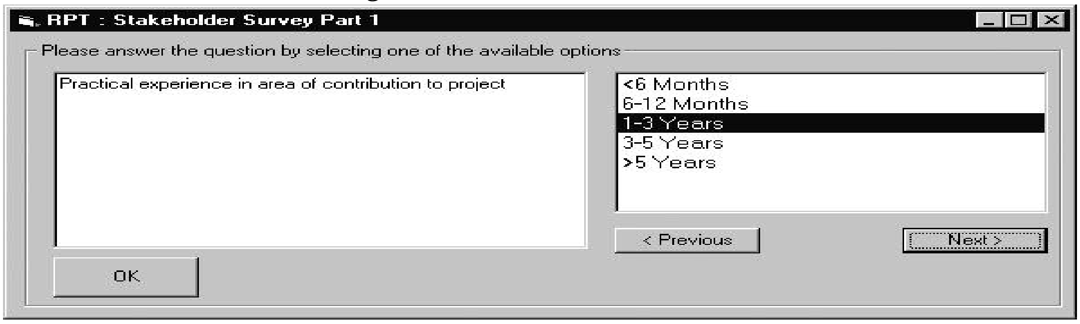


Figure 2. Stakeholder completes a questionnaire that is used to form the stakeholder profile model and ratings.

graphical rating scale which doesn't restrict the stakeholders to a strict ordinal scale [20]

- Rating the requirements based on objective measures such as usage metrics [21],
- Finding the dependencies amongst the requirements, and clustering the requirements with their dependencso as to achieve more efficient priorities.
- Using risk analysis techniques from Social Network Analysis (SNA) [22] to detect cliques amongst the stakeholders.

The Requirements Prioritisation Tool (RPT) consists of:

Stakeholder requirements and business goal ratings: Each stakeholder rates the importance of each requirement and business goal using the graphical fuzzy rating scale [21]. The stakeholder plots their point of preference on a line between the two extreme poles "absolutely not important" and " absolutely very important" and the middle point "moderately important". Then they are asked to draw a line to the left and right of the point signifying their range of acceptance or tolerance towards the two poles. The range and point are used to calculate the expected value of the

A stakeholder profile model: Each stakeholder is asked to complete a questionnaire that provides the tool with information about the stakeholder's role in the project, their skills, their experience in their specialised field, and their opinion on the benefits and consequences the introduction of the proposed product will have to their function within the company. The stakeholder is also asked to rate their knowledge on various non-functional issues. This information is used to formulate a stakeholder model for each stakeholder and to give each stakeholder a rating as to how important their opinion is to the project at hand. Each stakeholder is also asked to rate the other stakeholders as to their importance to the project at hand. stakeholder's preference using fuzzy logic techniques.

Influence factors: Each stakeholder rates how negatively or positively he believes each of the business goals influences each of the requirements using the graphical fuzzy rating scale. The stakeholder can express no opinion if they can't answer or don't want to answer the question. When a business goal can have both a negative and positive effect on a particular requirement then the stakeholder can simply put the red lines on the two poles and place the initial preference in the middle. This will produce a neutral answer.

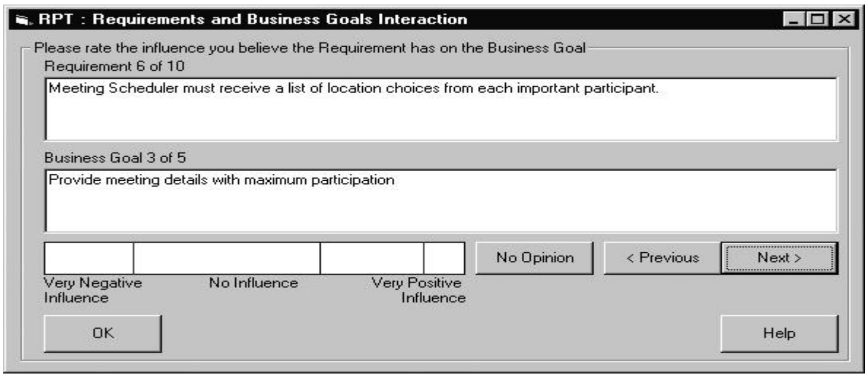
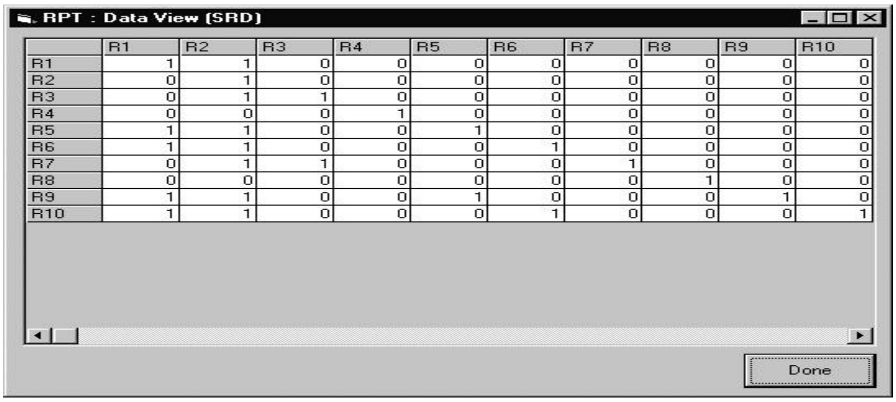


Figure 3. Stakeholders are asked to rate the degree of influence each requirement has on each of the business goals by using the graphical fuzzy rating scale.

Set of requirement dependencies: The dependencies can be detected from mapping the pre and post conditions from the whole set of requirements, based on the contents of each requirement statement. The resulting set of dependencies can be used to cluster the requirements. RPT produces a dependency matrix which is used to cluster the requirements in the prioritisation process.



	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10
R1	1	1	0	0	0	0	0	0	0	0
R2	0	1	0	0	0	0	0	0	0	0
R3	0	1	1	0	0	0	0	0	0	0
R4	0	0	0	1	0	0	0	0	0	0
R5	1	1	0	0	1	0	0	0	0	0
R6	1	1	0	0	0	1	0	0	0	0
R7	0	1	1	0	0	0	1	0	0	0
R8	0	0	0	0	0	0	0	1	0	0
R9	1	1	0	0	1	0	0	0	1	0
R10	1	1	0	0	0	1	0	0	0	1

Figure 4. A dependency matrix indicating dependencies amongst requirements. For example R5 is dependent upon R1 and R2.

Usage metrics: An objective measure developed from calculating the frequency of objects and actors used in a requirement statement or a use case [21]. The following values for object and actor usage for each use case (U_k^i and U_j^i) can be used to rank a set of use cases or requirements and produce an objective ranking for each use case or requirement U_i.

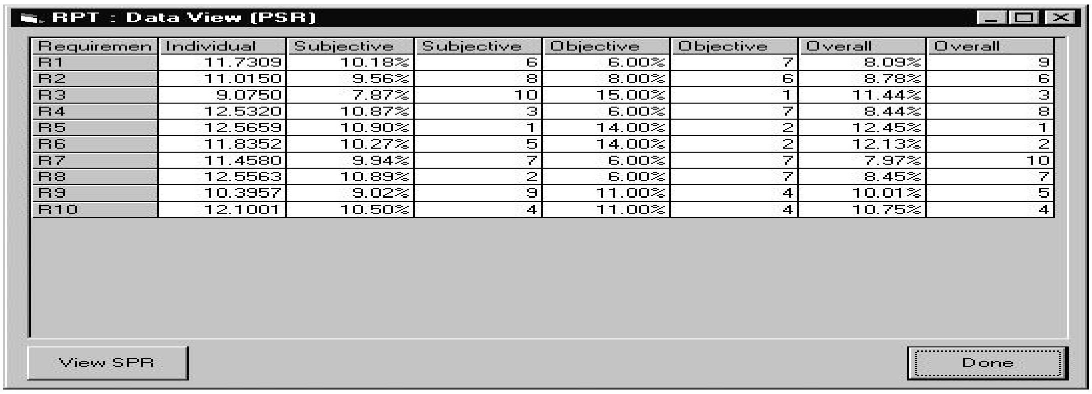
- **Object usage** for a particular use case of the set of use cases**:** $U_k^i = \Sigma^{n}_{k=1} f(X'_k)^i$ where k = number of the object, n = total number of objects, i = the number of the particular use case of set of use case, and X'_k = the frequency rating of a particular object in the use cases.
- **Actor usage** for a particular use case of the set of use cases**:** $U_j^i = \Sigma^{m}_{j=1} f(Y'_j)^i$ where j = number of the actor, m = total number of actors, i = the number of the particular use case of set of use cases, and Y'_j = the frequency rating of a particular actor in the use cases.

The subjective overall rating for each requirement is calculated from the formula:

$$R_i = \sum_{s=1}^{N} \sum_{b=1}^{M} S_s P_i I_{ib} B_b$$

where i = specific requirement, R_i = Requirement subjective overall rating, P_i = Stakeholder s rating for Requirement i, S_s = Stakeholder Profile rating for stakeholder s, I_{ib} = Influence rating of Requirement i on Business Goal b according to stakeholder s, B_b = Stakeholder s Rating of Business goal b, M = total number of business goals, N = total number of stakeholders, I = total number of requirements.

Overall rating of the requirement: The overall rating of the requirements is calculated from the average of the objective and subjective ratings for each requirement. The subjective and objective

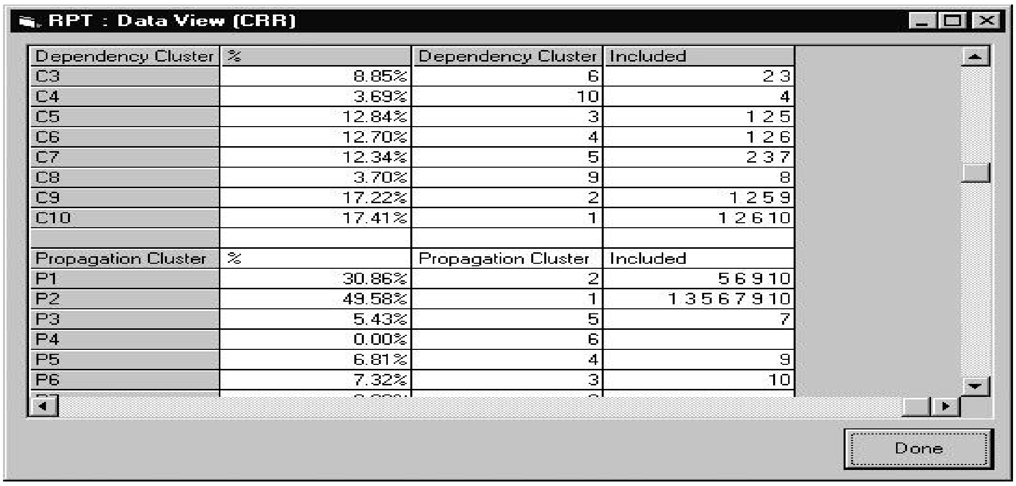


Requiremen	Individual	Subjective	Subjective	Objective	Objective	Overall	Overall
R1	11.7309	10.18%	6	6.00%	7	8.09%	9
R2	11.0150	9.56%	8	8.00%	6	8.78%	6
R3	9.0750	7.87%	10	15.00%	1	11.44%	3
R4	12.5320	10.87%	3	6.00%	7	8.44%	8
R5	12.5659	10.90%	1	14.00%	2	12.45%	1
R6	11.8352	10.27%	5	14.00%	2	12.13%	2
R7	11.4580	9.94%	7	6.00%	7	7.97%	10
R8	12.5563	10.89%	2	6.00%	7	8.45%	7
R9	10.3957	9.02%	9	11.00%	4	10.01%	5
R10	12.1001	10.50%	4	11.00%	4	10.75%	4

Figure 5. The subjective and objective ratings and rankings are used to calculate the overall prioritised ratings and ranks for the set of requirements.

Clustering the Requirements: Clustering the requirements is based on the dependency matrix. Ratings for each cluster are produced by using the aggregate of the individual ratings. The ratings are normalised and the priorities for the interdependent clusters of requirements are produced.

RPT can also produce propagational clusters from parent requirements. RPT detects which requirements form the basis for the most important requirements. By using the ratings for each requirement, the most important ancestors can be detected. In fact the dependencies of a set of requirements can be represented in the form of a binary matrix or with a tree structure.



Dependency Cluster	%	Dependency Cluster	Included
C3	8.85%	6	2 3
C4	3.69%	10	4
C5	12.84%	3	1 2 5
C6	12.70%	4	1 2 6
C7	12.34%	5	2 3 7
C8	3.70%	9	8
C9	17.22%	2	1 2 5 9
C10	17.41%	1	1 2 6 10
Propagation Cluster	%	Propagation Cluster	Included
P1	30.86%	2	5 6 9 10
P2	49.58%	1	1 3 5 6 7 9 10
P3	5.43%	5	7
P4	0.00%	6	
P5	6.81%	4	9
P6	7.32%	3	10

Figure 6. The dependency clusters (the groups of interdependent requirements) and the propagation clusters (requirements stemming from an ancestor) are ranked.

Risk Analysis Measures (RAM): RAM compares the individual stakeholder inputs or subjective ratings to the final priorities produced for the requirements and business goals. RAM also detects cliques amongst stakeholders in regards to stakeholder importance ratings, or in regards to the requirement, business goal, or influence ratings. RAM can detect agreement in preferences amongst subsets of stakeholders. These groupings are called cliques. These cliques can reveal biases and hidden agendas for a certain group of stakeholders.

RAM can also produce the synergy between the subsets of stakeholders in regards to their mutual preferences. It can show how interconnected a subset of stakeholders is in regards to their preferences or dislikes.

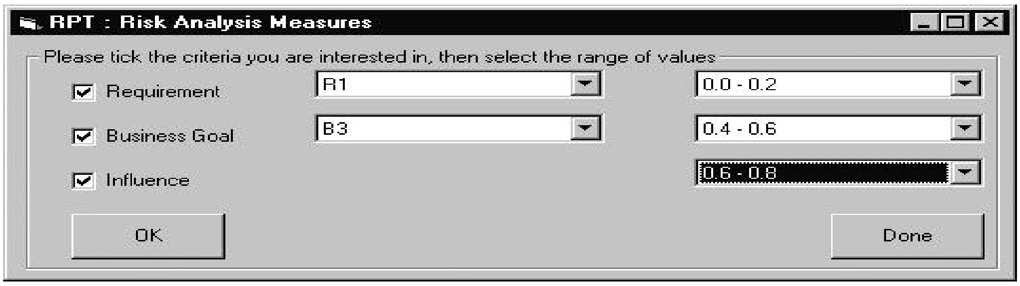


Figure 7. RAM detecting similarities in stakeholder preferences for the subjective ratings.

Stakeholder Association Measures (SAM) SAM applies the Spearman Rank-Order Correlation Coefficient [23] to provide an association percentage between an individual stakeholder's ratings on either the requirements or business goals, and the overall ratings. The association measure will indicate how much an individual stakeholder's ratings compares to the overall ratings.

SAM also produces the standard deviation amongst the stakeholders' opinions towards the importance of the requirements and business goals, compared to the overall ratings. SAM indicates on a bar chart the agreement and disagreement between an individual's ratings for the importance of the requirements and business goals compared to the overall ratings.

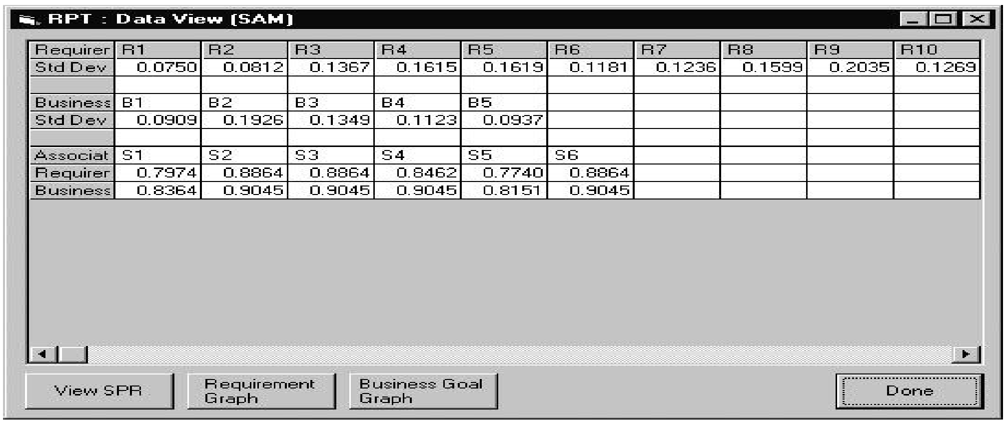


Figure 8. The standard deviation of the subjective input for the requirements and business goals, and the association measures amongst the stakeholder for their subjective ratings on the requirements and the business goals.

The Dilemma of Using Scales

Psychologists [16, 18] state that people's attitudes are mentally set in ranges rather than in single choices. Mentally people conceive attitudes and preferences as ranges of acceptance, ranges of tolerance, not as a single point. The notion of latitudes of acceptance and rejection in attitude measurement seems a more accurate representation of the mental attitudes. For instance, psychologists [16, 19] graphically portrayed the attitudes of people in relation to pacificism and militarism. The range of attitudes was rarely a single point, but rather a range extending between two points on the scale, which represented the range of opinion an individual would endorse.

In psychology and marketing, there is a proven mismatch between natural responses and the number of response categories used in a response scale. It is very difficult to predict the optimal number of response categories needed to elicit all the ranges of natural choices. There can be loss of information if a scale is too coarse, while on the other hand, a scale can become too complicated for the respondent if the scale is too fine. For example, hours of TV watched daily can be rated on a scale of (<1, 1-2, 2-3, 3+ hours). What does one do with someone who watches 1.5-2.5 hours of TV daily, or for the person who watches 20 minutes on Monday-Friday but at least 4 hours on the weekend?

Category descriptors can be misleading, and misunderstood. For example, if (High >25, Medium= 15-25, Low <15), 24 would be considered medium, even though it is only one point less than 25, which is high. Therefore, the scale indicates that one point is the true difference between the two categories (high, medium). Category descriptors are generally misleading and do not allow for gradation of membership. Single category responses do lead to a loss of information. An option is to use degrees of membership of items in each category so as to eliminate the potential loss.

The fuzzy graphic rating scale [21]offers a direct way of addressing the notion of ranges within the domain of attitude measurement. One can allow the person to express a tolerable or acceptable range of their preferences between two choices. The expected value of the fuzzy variable (the mental attitude) can then be calculated. For instance, one may ask the stakeholders to rate a particular requirement between the two poles of "not essential " and "essential".

Comparison of Framework and RPT with Other Methods RPT and its framework handle the following issues in requirements prioritisation that are not adequately handled in the other approaches:

- RPT provides a means to incorporate the business objectives of developing a system into the incremental development of the requirements.
- RPT allows business goals and stakeholder viewpoints to drive the evolution of requirements.
- RPT provides a means to incorporate the notion of dependencies amongst requirements in the prioritisation process.
- RPT provides a means to elicit accurate mental perceptions on stakeholder viewpoints by using the graphical fuzzy rating scale.
- RPT provides a means to detect cliques and discrepancies amongst the stakeholder opinions.
- RPT can be re-used by simply changing the set of requirements and asking the stakeholders to state their opinions on the new requirements and the influence they have on the business goals.

VALIDATION OF FRAMEWORK AND TOOL FOR PRIORITISING REQUIREMENTS

The RPT tool and framework was validated by applying the tool to various case studies. Each group had a set of high-level requirements and a set of high-level business goals to work with, and were asked to use conventional prioritisation techniques to prioritise the requirements, as well as using RPT. The groups consisted of stakeholders with specific roles and agendas. There were conflicting expectations and business goals amongst the group as well as cliques towards certain requirements, features, and business goals. The groups had to use the conventional techniques to produce a prioritised list of requirements based on the information and roles they were playing. They had to also use the RPT tool and framework to prioritise the same set of requirements and business goals. After the stakeholders used RPT, the requirements analyst viewed the results from the tool. The requirements analyst was able to detect the weaknesses in the group. The analyst was able to use the information to question certain stakeholder input and to stimulate further negotiations over which requirements to discard and which requirements to focus on in more detail.

The final reports revealed that the requirements analysts found it useful in assisting them in their elicitation and analysis of the requirements and in detecting potential problems with stakeholders. The customers and users found that RPT helped them solve conflicts in their group discussions, and bring hidden stakeholder agendas to light

CONCLUSION

RPT and the framework for prioritising requirements endeavour to prioritise high-level requirements with a set of high-level business goals**.** The objective of RPT is to assist the requirements engineer to prioritise a set of requirements early on in the systems development life cycle. It incorporates the multi-faceted issues of prioritising requirements and endeavours to assist in guiding the evolution of requirements from high-level descriptions to detailed requirements that will produce successful systems.

REFERENCES

1. Yourdon, E., *Death March Projects.* 1997: Prentice Hall.
2. Lubars, M., Potts, C., Richter, C. *A Review of the State of the Practice in Requirements Modeling.* in *Proceedings of the IEEE International Symposium of Requirements Engineering.* 1993: IEEE Computer Society Press.
3. Siddiqi, J., Shekaran, M.1996, Requirements Engineering: The Emerging Wisdom. *IEEE Software, p. 15-19,*March 1996
4. Christel, M.G., Kang, K. C., *Issues in Requirements Elicitation*., Technical Report CMU/SEI 12, Software Engineering Institute, Carnegie Mellon University, 1992.
5. Karlsson, J., Olsson, S., Ryan, K.1997, Improved Practical Support for Large-scale Requirements Prioritizing,. *Journal of Requirements Engineering*, Springer-Verlag London Limited,,. **2**: p. 51-60. 1997
6. Zowghi, D., Offen, R. *A Logical Framework for Modeling and Reasoning about the Evolution of Requirements.* in *International Symposium of Requirements Engineering.* 1997.
7. Akao, Y., *Quality Function Deployment: Intergrating Customer Requirements into Product Design*. Productivity Press, 1988
8. Saaty, L., *The Analytical Hierarchy Process*. New York: McGraw-Hill, 1980.
9. Karlsson, J., Ryan, K.1997, A Cost-Value Approach for Prioritizing Requirements. *IEEE Software*, **14**(5): p. 67-74, 1997.
10. Wiegers, K., *Software Requirements*. Microsoft Press, 1999
11. Moisiadis, F. 2001, 'A Framework for Prioritising Software Requirement.' in *Doctoral Workshop at the 5th International Symposium on Requirements Engineering (RE 2001)*. Toronto, Canada, 2001
12. Lamia, W. 'Integrating QFD with Object-Oriented Software Design Methodologies.' in *Transactions from the Seventh Symposium on Quality Function Deployment.* QFD Institute, 1995
13. Fehin, P., *Using QFD as a Framework for a User-Driven Participatory Design Process*, Ireland: Digital Equipment International BV.
14. Saaty, T., Kearns, K., *Analytical Planning: The Organization of Systems*. Pergamon Press, 1985
15. Moisiadis, F. 1999 'Case Study on the Use of Scaling Methods for Prioritising Requirements. in *9th Annual International Symposium of the International Council onSystems Engineering (INCOSE 1999)*. Brighton, United Kingdom. 1999.
16. Hesketh, B., Pryor, R., Gleitzman, M., and Hesketh,T., *Practical Applications*

and Psychometric Evaluation of a Computerised Fuzzy Graphical Rating Scale, in *Fuzzy Sets in Psychology*, T. Zetenyi, Editor, Elsevier Science Publishers B.V.: North-Holland 1988.
17. Cox III, E.P.1980, 'The Optimal Number of Response Alternatives for a Scale: A Review.' *Journal of Marketing Research*, American Marketing Association, 17: p. 407-422, 1980.
18. Viswanathan, M., Sudman, S., Johnson, M., 'Maximum Versus Meaningful Discrimination in Scale Response: A Perspective on the Optimal Number of Response Categories to Use in a Scale.' *Psychology and Marketing*, 1998.
19. Thurstone, L., 'Attitudes Can Be Measured.' *American Journal of Psychology*, 33: p. 529-554, 1922.
20. Moisiadis, F. 1999 'Improving the Scales Used in AHP for QFD'. *in 11th Symposium on Quality Function Deployment..* Detroit, USA: QFD Institute, 1999
21. Moisiadis, F. 2000, 'Prioritising Scenario Evolution'. in *International Conference on Requirements Engineering (ICRE 2000).* Chicago, USA: IEEE Computer Society, 2000.
22. Scott, J., *Social Network Analysis: A Handbook*. SAGE Publications, 1991.
23. Siegel, S., Castellan, N.Jr., *Nonparametric Statistics for the Behavioral Sciences*. 1988: McGraw-Hill.

ABOUT THE AUTHOR

Author name: Frank Moisiadis
Business affiliation: Macquarie University
Address: Computing Department
Macquarie University NSW 2109 Australia.
Phone number and fax:
Phone: +61 2 9850-9521
Fax: +61 3 9850-9551
E-mail: fmoisiad@ics.mq.edu.au

Frank Moisiadis is an Associate Lecturer at Macquarie University, Australia. His PhD research, funded by CSIRO-JRCASE, has been in requirements engineering, specifically in the elicitation and prioritisation of requirements. He has been the workshop organiser for the Doctoral Consortium for CAiSE 1999, and on the organising committee for ACRE 1999 and AWRE 2000. Recent publications include "Prioritising Software Requirements"(SERP 2002), "Prioritising Scenario Evolution" (ICRE 2000), "Prioritising Use Cases and Scenarios" (TOOLS Pacific 2000), and "Improving the Scales Used in AHP for QFD" (QFD 1999).

Keywords
Requirements Prioritisation, QFD, AHP, Stakeholder modelling, Requirements negotiations, Social Network Analysis.