The Utility Space Of Requirements Engineering

C.W. Johnson,
Department of Computing Science, University of Glasgow,
Glasgow, United Kingdom, G12 8QQ.
E-mail: johnson@dcs.glasgow.ac.uk,
Telephone: +44 41 330 8855, Fax: +44 41 330 4913.

ABSTRACT

This paper exploits the micro-economic concept of expected utility to characterise the costs and benefits of different levels of expenditure on requirements engineering. It is argued that different groups associate different levels of expected utility with particular development techniques. For instance, the advocates of formal methods argue that the high costs of mathematical specifications offer substantial benefits during the later stages of development. For other groups within the requirements engineering community, the high costs of formal specification techniques are perceived to offer few rewards. This paper argues that utility curves can be exploited to characterise these different viewpoints. The intention is not to map out all of the complex factors that affect human decision making but to focus on the interaction between issues of cost and probability in commercial project management. The resulting models are then used to identify an agenda for progress in the field of requirements engineering. Quality assessment procedures must be developed to demonstrate the benefits of particular approaches. Cost effective methods must be developed to encourage the application of advanced techniques. Finally, it is argued that the application of high-cost approaches must be targeted so that they produce the greatest expected utility.

Problem area: 1.6, 1.5. Contribution: B, C. Word count: 4,581.

1 Introduction

Requirements engineering involves the identification, specification and evaluation of design objectives. Recent years have seen considerable progress in each of these areas. Techniques have been developed to identify natural language requirements from customers, operators and regulatory authorities. Precise specifications can be derived from vernacular requirements. Techniques have been developed to reason about requirements; to prove their consistency and demonstrate completeness. Abstract specifications can be refined towards final implementation. In spite of this progress, there seem to be a number of barriers to the widespread application of advanced requirements enginering techniques. For instance, some companies are reluctant to exploit formal methods [9] and ethnomethodology [11]. This skepticism stems from the high costs and low benefits that are anticipated from these 'advanced' techniques.

The following pages argue that expected utility can be used to characterise different attitudes towards the costs and benefits of requirements engineering. Section 2 introduces the empirical and theoretical foundations of utility. A number of alternative approaches to the application of this concept are discussed. Generic models of utility are preferred to the context-dependent models that can be derived from laboratory studies. Section 3 identifies four different utility curves that capture attitudes towards the costs and benefits of requirements engineering. These are: the traditional; the constant marginal

utility; the high cost and the critical mass models of utility. They do not represent all of the possible attitudes towards requirements engineering. They do, however, represent common attitudes towards this development activity. Section 5 exploits this analysis to examine ways in which the field of requirements engineering might progress in the next five years. In particular, it is argued that quality assessment procedures are required to promote good practice and increase the anticipated utility of requirements engineering. Low cost techniques must also be developed to increase the commercial exploitation of research in this area. Section 6 presents the conclusions that can be drawn from this research.

2 What Is Utility?

Requirements engineering consumes resources. Finite funds and skills are directed towards those activities that are perceived to provide the greatest benefit during development. It is, therefore, appropriate to consider the relationship between the resources allocated to requirements engineering and the perceived benefits of this activity. This paper exploits the economic concept of expected utility to analyse this relationship. Utility has a long history. It has been used by philosophers and political economists to explain the phenomenon of value [3]. It can be thought of as a measure of 'desire' or the capacity of a good or service to satisfy a need. It is a subjective concept, rather than a measure of fitness for purpose. In the context of this paper, utility is the perceived benefit that system developers associate with requirements engineering. For instance, a developer might hold the view that requirements engineering has a relatively low utility when compared with implementation. Such attitudes have important consequences for the field of requirements engineering. If an individual holds this view then they are unlikely to exploit many of the high cost techniques being advocated by research in this area. It is important to understand the rational foundations that support such attitudes if these perceptions are to be altered.

A number of different approaches might be used to represent the expected utility of requirements engineering. The benefits of this activity can be expressed in terms of the monetary savings that are anticipated through reduced re-development costs. Unfortunately, these savings cannot easily be quantified. It is difficult to estimate the monetary value of improved safety. Alternatively, empirical techniques can be applied to determine the relative utility that an individual perceives between competing goods and services. These approaches do not attempt to quantify the monetary benefits of an activity, such as requirements engineering. Instead, they identify subjective preferences. For instance, March and Simon's 'satisficing' can be used to iteratively refine the constraining equations that characterise preferences between two development tasks [21]. An individual might be asked to indicate a desired allocation of resources between requirements engineering and implementation. An interviewer then points out any constraints that affect this decision, such as legislative obligations during particular stages of requirements engineering. The developer must then decide whether to allocate more or less resources to requirements engineering. This iterative exchange continues until the developer is satisfied with the trade-off between the two activities.

A number of limitations restrict the application of the procedure described in the previous paragraph. The interview process can yield biased results. They capture an individual's preferences at a particular moment in time. Their expressed preferences may not be maintained under the deadlines and pressures of a working environment. In contrast, this paper develops generic models of utility. The intention is to characterise viewpoints that are common amongst the many different groups which are involved in requirements engineering. This raises the question of validation. Without empirical evidence, it is difficult to determine whether our models actually capture widely held attitudes towards the costs and benefits of requirements engineering. A number of surveys into industrial requirements engineering have been used to avoid this criticism [2, 9, 11, 20, 22]. Many of these sources adopt particular perspectives upon development. These viewpoints have also been characterised in terms of utility curves, for instance see figure 3.

3 Under Investment in Requirements Engineering

Boehm argues that only 6% of project costs are allocated towards requirements engineering [5]. As a consequence, the constraints that a system must satisfy are often picked up late in development. Fixes are introduced by increasing the allocation of funds to re-design and re-implementation. Between 40% and 75% of development costs may be consumed in repairing problems that were missed during requirements analysis [6]. Figure 1 captures the viewpoint that leads to such problems; requirements engineering is treated as a low cost process that requires relatively few resources. In this diagram we assume the existence of a function, U, that returns the expected utility gained by a particular level of resource allocation for a particular development activity. The marginal utility, or the amount of value associated

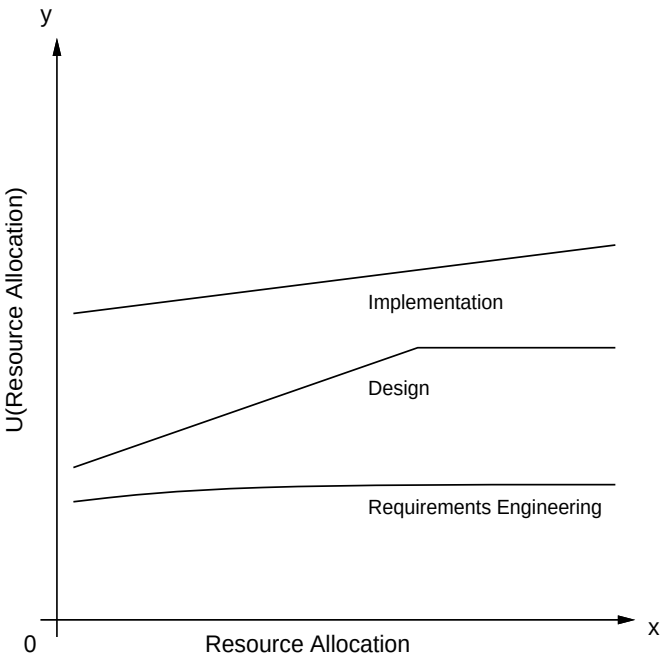

Figure 1: Comparative Utility Of Development

with an additional unit of resource, changes little in this view of requirements engineering. Beyond the initial allocation to this activity, little is to be gained from allocating additional resources. This contrasts with design, there is a high marginal utility up to a point where a plateau is reached. This indicates the position at which the allocation of further resources is not perceived to yield any additional rewards. Finally, implementation is shown to have a positive marginal utility at each level of resource allocation. In this viewpoint, only a minimal allocation would be made to requirements engineering. A common assumption in decision theory is that 'rational agents' attempt to maximise their total expected utility. In figure 1, a developer would attempt to allocate as many resources as possible towards implementation. The relative expected utility of this activity would outweigh the high marginal utility of design. This is an extreme viewpoint. It is important to re-iterate at this point that utility is a measure of 'desire' and not fitness for purpose. It is also important, to identify the reasons that lie behind this view of requirements engineering. Lubars, Potts and Richter provide the best analysis of this in their study of ten U.S. corporations [20]. One of the reasons they cite is the under-capitalisation of requirements

activities. They argue that requirements engineering suffers because it is not supported by adequate resources, such as CASE tools. They also mention the reluctance of organisations to re-train developers in new techniques; "organisations equate requirements analysis with structured analysis". This suggests that there may be a circular problem. Resources are not allocated to requirements engineering because it has a relatively low expected utility. Requirements engineering has a relatively low expected utility because it is not attracting sufficient resources to be effective.

4 High Utility Models Of Requirements Engineering

The previous section has identified under-investment as the major problem facing requirements engineering. This has been explained in terms of the low expected utility that is frequently associated with this development activity. In contrast, this section analyses viewpoints that associate relatively high levels of utility with requirements engineering.

4.1 The Constant Marginal Utility Model Of Requirements Engineering

The low marginal utility shown in figure 1 represents an extremely pessimistic view about the benefits that might accrue from the allocation of resources to requirements engineering. Figure 2 presents an alternative viewpoint. Each additional resource spent on requirements engineering results in the same increase in total utility. In other words, there is a constant positive marginal utility. The industrial perception of requirements engineering, identified by Lubars, Potts and Richter [20], contrasts markedly with the view represented by figure 2. Boehm provides an earlier set of figures than that cited in the previous section

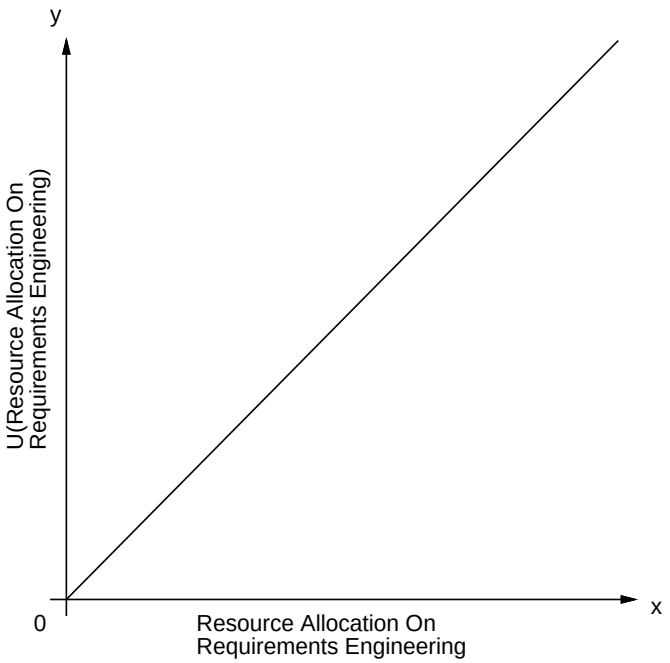

Figure 2: Constant Marginal Utility View Of Requirements Engineering.

which indicate that this model may represent commercial attitudes to requirements engineering [4]. He

estimates that requirements and design consume between 33% and 46% of total development resources [4]. From these figures it would appear that industry is attempting to maximise its utility by allocating extremely large amounts of resources to the early stages of development. Unfortunately, further analysis of Boehm's figures reveals that this is not the case. The majority of the percentage cited above is allocated to the detailed design of a system and not to requirements engineering. The curve illustrated in figure 2 does, however, capture the attitude of some researchers in this field. Simple two or three stage process models [10] have, by successive accretions, developed into twelve stage models [24]. Such approaches have introduced more and more activities into the early stages of development. They assume that the consumption of additional resources will provide corresponding benefits for systems development.

The viewpoint captured by figure 2 represents over-optimistic expectations. It is unlikely that there will be a constant positive marginal utility. The benefits brought by each additional resource will change at different levels of total expenditure. Initially, small increments to the overall requirements budget will have little impact. As the level of expenditure rises then developers might expect that the effect of each additional resource will also increase. This viewpoint can be seen in Hawley, de Pontbriand and Frederickson's analysis of the acquisition process for the U.S. Army's Forward Area Air Defence System [13]. They emphasise the need for "overage" during the early stages of development. By this they mean that developers should always over estimate the requirements budget. The marginal resources allocated at this high level of expenditure are assumed to provide high levels of utility for the overall development process. This viewpoint is captured by the high cost model of requirements engineering.

4.2 The High Cost Model Of Requirements Engineering

Figure 3 represents the view that additional expenditure on requirements engineering is expected to bring increased benefits at high levels of resource allocation. Expenditure at low levels of allocation is perceived to have less of an effect upon development. This model characterises the viewpoint of many in the requirements engineering community. For instance, elicitation techniques that exploit ethnomethodology or non-intrusive video recordings require considerable expenditure in order to be effective [11]. They require considerable skill on the part of the analyst and they require large amounts of time to collate the results. Similarly, the use of formal methods requires high levels of resource allocation. The application of discrete maths and logic demands significant skill on the part of the developer [14, 17]. If a limited resource allocation results in inadequate training then mathematical notations may be abused. Specifications may be inconsistent or incomplete. Significant amounts of time and money must be used to avoid these problems. For instance, the proofs conducted on the SACEM project for the Paris subway took 32% of initial development costs. This relatively high level of resource allocation still produced high levels of consumer satisfaction because "data suggests that the additional activities required for assimilating and using formal methods did not add to the overall process costs" [9]. The high level of expenditure on requirements engineering was outweighed by the saving that were observed during the certification process.

The utility of curve of figure 3 represents the perception of some sectors within safety critical industries. It does not reflect the perceptions of other industries. Sommerville argues that software management "is unwilling to adopt new techniques whose payoff is not obvious. It is hard to demonstrate that the relatively high cost of developing a formal system specification will reduce overall software development costs" [26]. It can, therefore, be argued that greater attention ought to be paid to lower cost approaches if mathematical specification techniques are to have a wider industrial application. Later sections will outline ways in which this might be done. For now it is sufficient to observe that there is a pressing need for better training and tool support in the large scale application of formal methods. The same survey that reported SACEM's success also concluded that more attention needed to be spent "on developing notations more suitable to use by individuals not expert in formal methods or mathematical logic" [9]. The intention would be to increase the expected utility of these techniques at lower levels of expenditure. This viewpoint is represented by the critical mass model of requirements engineering.

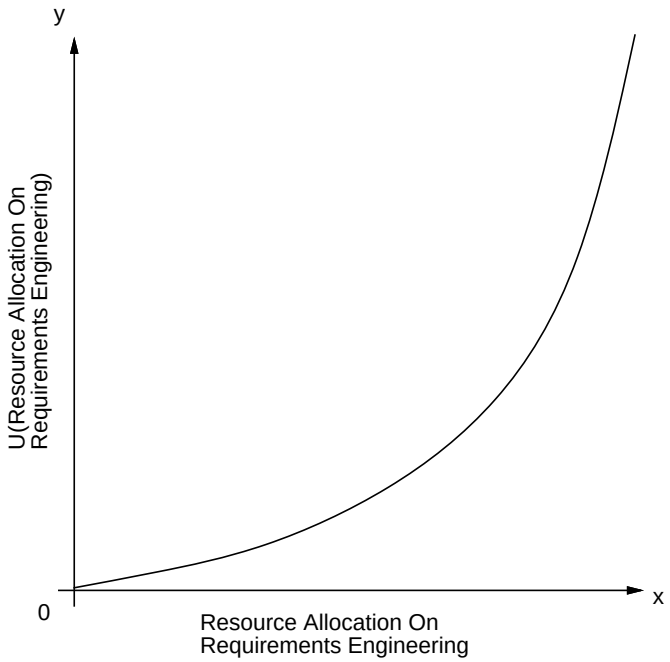

Figure 3: High Cost View Of Requirement Engineering.

4.3 The Critical Mass Model Of Requirements Engineering

Developers might expect a positive increase in utility as each new member of staff is added to a requirements team. In the real world, however, things do not always work out as intended. It is unlikely that the continued allocation of time and money will provide the benefits that are anticipated in high cost approaches to requirements engineering. Larger requirements teams suffer from increasing communications problems [18]. Clegg and Wall argue that 'political' and managerial problems eventually inhibit development [8]. Groups can be burdened by the additional demands of training new members [7]. Experience, therefore, suggests that diminishing marginal returns limit the growth in expected utility. In other words, the allocation of additional resources will result in smaller increases in expected utility. This is characterised by the graph shown in figure 4. In an extended form of this model, the allocation of increased resources may eventually have a negative effect on overall utility.

The critical mass model represents the viewpoint of those who advocate low cost approaches to requirements engineering. The use of introspection, questionnaires and interviews can quickly yield benefits for a minimal investment [1]. In other words, the critical point at which resources are anticipated to yield significant benefits occurs at a much lower level than in the high cost model. More expensive techniques are perceived to provide only negligible benefits beyond those offered by initial investments. There are many important consequences for the field of requirements engineering if figure 4 accurately reflects commercial perceptions. For instance, Sommerville, Rodden, Sawyer and Bentley have investigated the technical problems of of exploiting ethnomethodology during systems development [27]. They conclude that "it will be many years before the participation of a sociologist in a requirements team is normal practice". Our analysis suggests that even if the technical problems can be resolved, there remains the problem of altering perceptions. Developers must be convinced that the costs associated with these approaches are justified by their potential benefits over low cost approaches.

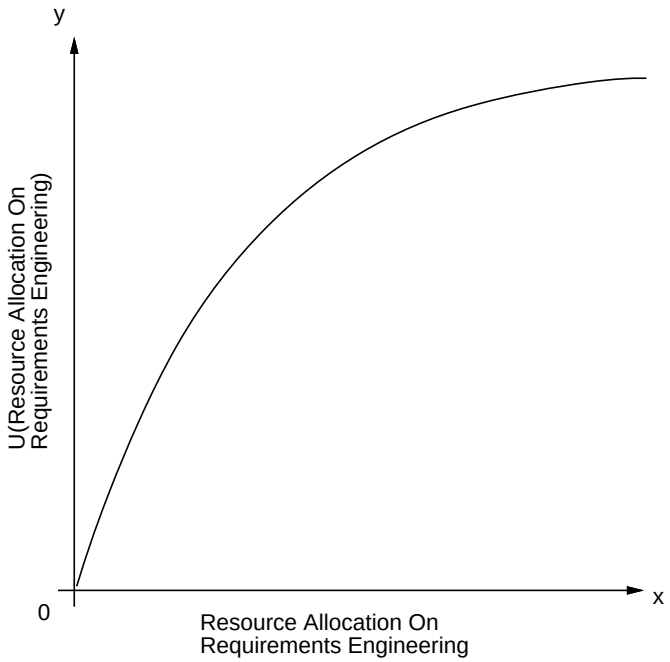

Figure 4: Critical Mass View Of Requirements Engineering.

5 Altering Perceptions Of Requirements Engineering

The following section identifies ways of altering the expected utility of requirements engineering. The first approach focuses upon technology transfer between academic research and commercial practice. It is argued that targeting techniques, such as Goguen and Linde's *zooming*, must be developed to improve the cost-effectiveness of advanced requirement engineering techniques. The second approach focuses upon quality assessment. It is argued that legislative and commercial controls can be applied to encourage good practice during the early stages of development.

5.1 Reducing Costs

Figure 3 characterised the high cost perception of requirements engineering. It was argued that many of the most advanced approaches, such as ethnomethodology and formal specification, are perceived to offer significant levels of utility at high levels of expenditure. It was also argued that even if the technical problems can be solved then it does not follow that these approaches will be adopted by industry. Research must increase the expected utility of these techniques at relatively low levels of expenditure. Such observations are far from novel. Craigen, Gerhart and Ralston insist on the need for improved tools to support the application of formal methods [9]. Wright and Monk argue for 'cost-effective' elicitation techniques to support video analysis [28]. The effects of these techniques can be illustrate by the curves shown in figure 5. The dotted line shows that the rise in utility occurs at lower levels of expenditure than the unbroken line which is taken from figure 3. Greater benefits are to be expected at lower levels of expenditure. This reflects the need to minimise the perceived costs of requirements engineering.

Decision theory suggests that there might be further benefits from the development of low cost tech-

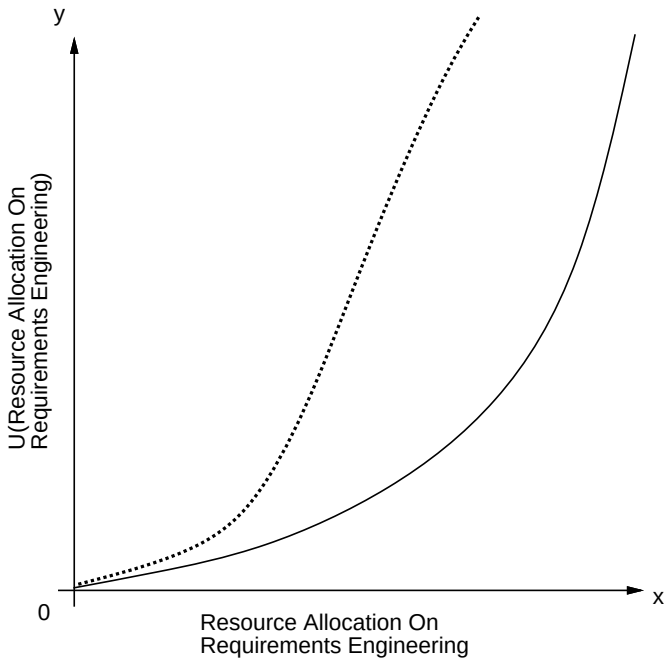

Figure 5: The Impact Of Low Cost Techniques.

niques. These stem from the commercial perception of risk. Von Neumann and Morganstern created the foundations for modern work in this area when they argued that the utility of an event multiplied by its probability gives its effective utility [23]. In other words, people make decisions by assessing the potential benefits of services and the likelihood that the resources which they allocate will actually deliver those services. In terms of this paper, developers must consider the likelihood that a particular level of investment in requirements engineering will actually result in the anticipated benefits. The high cost utility model, shown as the unbroken line in figure 5, represents an inclination to take risks. Economic theory defines the risk preference coefficient to be the second derivative divided by the first [12]. Rather than invest the large amounts of resources that are expected to provide high levels of utility, developers might be tempted to gamble by reducing their levels of expenditure. The rise in marginal utility in the low cost approach, shown as the broken line of figure 5, is more evenly distributed. It is less concave than the high cost model. From this it follows that low cost development techniques will not only be justified in terms of the absolute expenditure required for requirements engineering but they will also reduce develops' inclination to take risks with that expenditure.

This analysis raises the practical question of how to reduce the costs of requirements engineering. The work cited in previous paragraphs reflects the current interest in tool support and improved elicitation methodologies. These approaches are unlikely to be sufficient to persuade industry of the benefits of advanced requirements techniques. The lack of appropriate skills will continue to limit the practical application of formal methods and ethnomethodology. This situation may not improve until industry begins to recruit personnel with relevant skills. Again, there is a circular problem. Advanced techniques will not be applied until enough practitioners have acquired the relevant skills. Practitioners will not acquire the relevant skills until their industrial benefits have been proven. Goguen and Linde identify a potential solution to this problem when they "recommend a *zooming* method of requirements elicitation,

whereby the more expensive but detailed methods are only employed selectively for problems that have been determined by other techniques to be especially important" [11]. A number of companion papers describe our attempts to apply this approach. Accident analyses [15] and probabilistic risk assessments [16] are being exploited to target the use of formal methods. Additional resources should be allocated to those areas of a system which have failed in the past or are considered likely to cause problems in the future. Many problems still remain. For instance, developers will still not recruit low cost requirements engineering techniques if they are not convinced of their benefits. Quality control procedures provide a potential solution to this problem.

5.2 Improving Quality

The traditional model of requirements engineering, in figure 1, has been used to identify the circular problem of under-investment in requirements engineering. Resources are not allocated to requirements engineering because it has a relatively low expected utility. Requirements engineering has a relatively low expected utility because it is not attracting sufficient resources in order to be effective. Regulatory and legislative bodies affect this perception. Developers can be forced to increase the level of resources that are allocated to requirements engineering. For instance, safety cases must be submitted during the early stages of development for systems that interact with 'major hazards' in the United Kingdom. These documents must convince the Health and Safety Executive that safety requirements have been identified. The U.K. Defence Standard 00-55 plays a similar role in the military acquisition process. A minimum level of requirements expenditure is enforced by the identification of approved development techniques, including the use of formal methods [22]. Regulation and legislation ensure that low levels of expenditure on requirements engineering result in very low levels of utility; proposed systems will be rejected as unsafe. Figure 6 illustrates the way in which regulatory authorities affect developers' attitudes

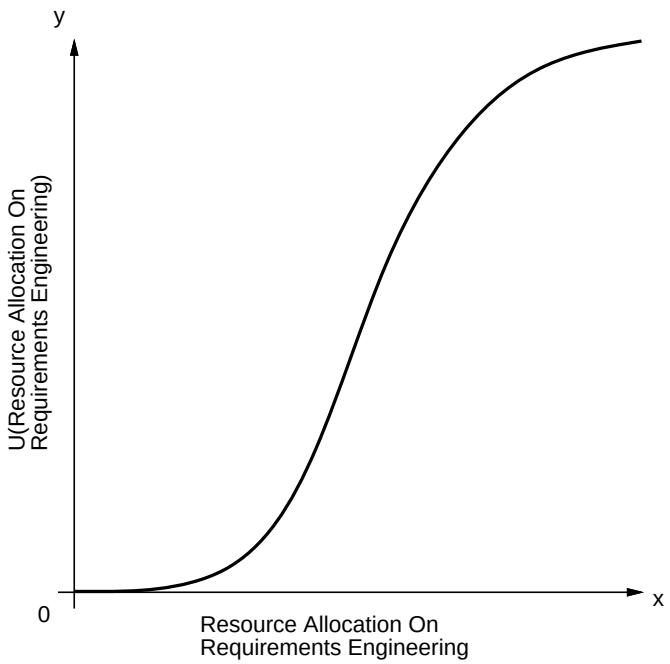

Figure 6: The Impact Of Legislation And Regulation.

towards requirements engineering. Zero utility is associated with low levels of expenditure. Small scale expenditure on requirements engineering is not perceived to provide any benefits; it will not satisfy the quality assessment procedures imposed by external authorities. This breaks the cycle of under-investment. Positive marginal returns yield a rapid increase in utility; in a similar manner to the second curve shown in figure 5. Eventually, diminishing marginal returns set in. At this point, the regulatory and legislative authorities are considered likely to accept the system. Additional resources need not, therefore, increase total utility.

It is not always possible for external agencies to impose minimum standards on requirements engineering. Practical and economic considerations prevent the use of regulatory authorities to monitor the development of every information processing system. There are, however, encouraging signs which indicate that this may not be necessary. Many companies are acting in a similar way to regulatory and legislative organisations. They have introduced assessment teams to monitor the products of each stage of development [25]. If requirements do not meet the standards imposed by these groups then they are returned for further consideration. Unfortunately, such practices are not widespread. There are many remaining research issues in this area. The techniques exploited by quality assessment groups often seem to be ad hoc and subjective. For instance, informal walkthroughs and peer reviews are commonly used to assess requirements documents. Clegg and Wall identify a number of problems with these techniques [8]. Peer reviews can be affected by 'political' pressures from within an organisation. More research is urgently needed into the organisational and technical problems of quality assessment for requirements engineering. Lee's work into quality circles for software engineering provides a useful starting point for this research [19].

6 Conclusion

This paper has argued that the expected utility of requirements engineering affects the level of resources that are allocated to this development activity. Four different views have been identified: the traditional; the constant marginal utility; the high cost and the critical mass. The traditional models represents requirements engineering as a low cost process that requires relatively few resources. This perspective leads to under-investment in requirements engineering. Additional resources are sucked into later stages of development to repair deficiencies that might have been identified earlier in the development of an application. This view contrasts with the constant marginal utility model. Each additional resource spent on requirements engineering results in the same increase in total utility. It seems likely, however, that marginal utility will vary at different levels of expenditure. This view has been represented by the high cost model. Additional expenditure on requirements engineering is expected to bring increased benefits at high levels of resource allocation. Many of the advanced techniques that are being proposed require significant resources if they are to be effective. Finally, this model has been contrasted with the critical mass utility curve. The allocation of additional resources will result in smaller increases in expected utility. High cost techniques are perceived to offer few benefits over an initial allocation on requirements engineering.

The utility models suggest a number of future directions for requirements engineering. The first of these has been identified from the formal relationship between utility and risk in decision theory. The risk preference coefficient of a utility curve is defined to be the second derivative divided by the first [12]. The high cost model represents an inclination to take risks with the allocation of resources towards requirements engineering. Rather than invest the large amounts of resources that are expected to provide high levels of utility, developers might be tempted to gamble by reducing their levels of expenditure. This contrasts with the critical mass model that is characteristic of risk aversion. The high utility of relatively low levels of expenditure will encourage developers not to risk the allocation of resources to this development activity. From this it follows that developers must be provided with low cost techniques for requirements engineering. This is not a novel observation; it lies behind research into cost effective

elicitation techniques [28] and work into tool support for formal methods [9]. Unfortunately, these tools and techniques still require considerable levels of skill and investment. It is has, therefore, been argued that targeting techniques, such as Goguen and Linde's *zooming*, must be developed to improve the cost-effectiveness of advanced requirement engineering methods. We are exploiting these techniques to focus the use of formal methods [15, 16].

The second direction for future work in requirements engineering builds upon the practices in safety-critical industries. Expected utility can be affected by the monitoring processes that are used to validate requirements documents. The intervention of legislative and regulatory bodies, such as the Health and Safety Executive, encourages the allocation of resources towards requirements engineering. These bodies ensure that low levels of expenditure result in very low levels of utility; proposed systems will be rejected as unsafe. It has been argued that other industries can achieve similar effects by introducing quality monitoring groups. This can only be successful if developers are provided with techniques that provide feedback on the quality of requirements documents. Recent investigations suggest that ad hoc management techniques and unstructured walkthroughs have a number of limitations [8, 19]. More research is urgently needed into into the organisational and technical problems of quality assessment for requirements engineering.

Acknowledgements

This work is funded by the United Kingdom's Science and Engineering Research Council, grant number GR/J07686. Many of the problems indentified in this paper are also apparent in other domains. Our analysis of utility curves has also been applied to commercial perceptions of interface developoment. It is apprent that many of the problems of technology transfer are common to both of these areas.

References

[1] D.C. Andrews. JAD: A crucial dimension for rapid applications development. *Journal Of Systems Management*, pages 23–31, March 1991.

[2] S. Austin and G.I. Parkin. Formal methods: A survey. Technical report, Division Of Information Technology And Computing, The National Physical Laboratory, Teddington, United Kingdom, 1993.

[3] R.D. Collinson Black. Utility. In M. Milgate J. Eatwell and P. Newman, editors, *Utility And Probability*, pages 295–302. Macmillan, London, United Kingdom, 1990.

[4] B.W. Boehm. The high cost of software. In E. Horowitz, editor, *Practical Strategies For Developing Large Software Systems*. Addison Wesley, Reading, Massachusetts, 1975.

[5] B.W. Boehm. *Software Engineering Economics*. Prentice Hall, London, United Kingdom, 1981.

[6] G. Booch. *Software Engineering With Ada*. Benjamin/Cummings, Wokingham, United Kingdom, 1987.

[7] F.P. Brooks. *The Mythical Man-Month: Essays On Software Engineering*. Addison Wesley, Reading, United States of America, 1982.

[8] C. Clegg and T.D. Wall. Managing factory automation. In F. Blackler and D. Obourne, editors, *Information Technology And People*, pages 45 – 64. British Psychological Society, Leicester, United Kingdom, 1987.

[9] D. Craigen, S. Gerhart, and T. Ralston. An international survey of industrial applications of formal methods. Technical Report NISTGCR 93/626, U.S. Department of Commerce, National Institute Of Standards And Technology, Githersburg, United States of America, 1993.

[10] A. Finkelstein, J. Kramer, and M. Goedicke. Viewpoint-oriented software development. In *Proceedings Of Le Génie Logiciel Et Ses Applications*, pages 337–351. Tolouse, 1990.

[11] J.A. Goguen and C. Linde. Techniques for requirements elicitation. In S. Fickas and A. Finkelstein, editors, *Requirements Engineering '93*, pages 152–163. IEEE Computer Society Press, 1993.

[12] B. Hansson. Risk aversion as a problem of conjoint measurement. In P. Gardenfors and N.E. Sahlin, editors, *Decision, Probability And Utility*, pages 136–158, Cambridge, United Kingdom, 1988. Cambridge University Press.

[13] J.K. Hawley, R.J. dePontbriand, and E.W. Frederickson. Making MANPRINT work: The lessons of the FAADS experience. In D.L. Pettigrew, editor, *Proceedings Of The 33rd Annual Meeting Of The Human Factors Society*, pages 1049–1053. Human Factors Society, Santa Monica, United States Of America, 1989.

[14] C.W. Johnson. A probabilistic logic for the development of safety-critical interactive systems. *International Journal Of Man-Machine Studies*, 39(2):333–351, 1993.

[15] C.W. Johnson. The formal analysis of human-computer interaction during accidents investigations. In G. Cockton, S.W. Draper, and G.R.S. Weir, editors, *People And Computers IX*. Cambridge University Press, Cambridge, United Kingdom, 1994.

[16] C.W. Johnson. The role of error-tolerant design in minimising the impact of risk. In I. Glendon and N. Stanton, editors, *Risk Assessment And Risk Reduction: A Conference In Memory Of Tom Hoyes*, Aston, United Kingdom, 1994. Aston University Business School.

[17] C.W. Johnson and M.D. Harrison. Using temporal logic to support the specification and prototyping of interactive control systems. *International Journal Of Man-Machine Studies*, 36:357–385, 1992.

[18] M. Klein. Capturing design rationale in concurrent engineering teams. *IEEE Computer*, 26(1):39–47, 1993.

[19] B.K. Lee. Implementing a quality circle programme for computer professionals. *Computer Systems Science and Engineering*, 1(1):65–67, 1985.

[20] M. Lubars, C. Potts, and C. Richter. A review of the state of the practice in requirements modelling. In S. Fickas and A. Finkelstein, editors, *Requirements Engineering '93*, pages 2–14. IEEE Computer Society Press, 1993.

[21] J.G. March and H.A. Simon. *Organisations*. Wiley, New York, United States of America, 1958.

[22] P. Mukherjee and V. Stavridou. The formal specification of safety requirements for storing explosives. *Formal Aspects Of Computing*, 5:299–336, 1993.

[23] J. Von Neumann and O. Morgenstern. *The Theory Of Games And Economic Behaviour*. Princeton University Press, Princeton, United States of America, 1944.

[24] W.E. Rzepka, J.L. Sidoran, and D.A. White. Requirements engineering technologies at Rome laboratory. In S. Fickas and A. Finkelstein, editors, *Requirements Engineering '93*, pages 15–18. IEEE Computer Society Press, New York, United States of America, 1993.

[25] A.P. Sage. Systems engineering and information technology: Catalysts for total quality in industry and education. *IEEE Transactions on Systems, Man and Cybernetics*, 22(5):633–664, 1992.

[26] I. Sommervile. *Software Engineering.* Addison Wesley, Reading, Massachusetts, 1990.

[27] I. Sommerville, T. Rodden, P. Sawyer, and R. Bentley. Sociologists can be surprisingly useful in interactive systems design. In A. Monk, D. Diaper, and M. Harrison, editors, *People And Computers VII: Proceedings Of HCI'92*, pages 341–353. Cambridge University Press, Cambridge, United Kingdom, 1992.

[28] P.C. Wright and A.F. Monk. A cost-effective evaluation method for use by designers. *International Journal of Man-Machine Studies*, 35(6):891–912, 1991.