Electronic version of an article published in **Gervasi, V.; Sawyer, P.; Paech, B. (Hrsg) Proceedings of the 12th International Working Conference on Requirements Engineering: Foundation for Software Quality (REFSQ'06), Reihe: Essener Informatik Beiträge Band 11, pp. 249-250**



http://www.refsq.org/

# Benefit Estimation of Requirements Based on a Utility Function

Andrea Herrmann, Barbara Paech

University of Heidelberg, 69120 Heidelberg, Germany
{herrmann, paech}@informatik.uni-heidelberg.de
http://www-swe.informatik.uni-heidelberg.de

Clearly defined requirement priorities are the essential basis for conflict solution among requirements, for decision-making during architectural design and for test case prioritization. Criteria for the prioritization of requirements can be their benefit for the stakeholders, cost and calendar time for realization, risk, penalty, urgency, importance of the stakeholder defining it, volatility, system impact, complexity, etc. [1-4].

In this work, we model the benefits of a system and of requirements by a utility function, like it is used in economics. We will show that such a function explains many experiences and results of other researchers about benefit estimation.

"In economics, utility is a measure of the happiness or satisfaction gained consuming good and services." [5] “In utility theory, a consumer has a utility function $U(x_i)$ where $x_i$ are amounts of goods with index i.” [6] Such a utility function $U(S_{xi})$ can also describe the total benefit of a system $S_{xi}$ which is designed to realize a subset of all requirements which the stakeholders defined. The variables $x_i$ describe whether a requirement is realized ($x_i$=1) or not realized ($x_i$=0). $U(S_{xi})$ is defined but unknown and must be estimated by the stakeholders. We want to mention that the utility function and consequently its estimation can be different for each stakeholder.

One can not expect the stakeholders to estimate the function $U(S_{xi})$ for all combinations of $x_i$. It is even difficult to estimate the total utility of a complex system. Instead, usually the utilities of single requirements or groups of requirements are determined. In terms of the utility function, requirement A´s utility is $u_A(S) = U(S_A) – U(S)$, where system $S_A$ is equal to S, except for requirement A being realized in $S_A$ and not in S. $u_A(S)$ depends on which requirements are realized by reference system S.

In terms of the utility function $U(S)$ and $u_A(S)$, some well-known observations during requirements prioritization can be described.

Several authors distinguish between the utility respectively satisfaction of the stakeholder when a certain requirement is realized and the dissatisfaction when it is not realized, and they find that they are not equal [7,8]. But: The dissatisfaction *d* experienced by the stakeholder when A is removed from $S_A$ is defined as $d_A(S_A) = U(S_A) – U(S)$ and should be equal to the satisfaction $u_A(S)$. So, why do stakeholders give different answers? Apart from misestimations and imagination problems, it might also happen that *u* and *d* are estimated relative to different reference systems. Fig. 1 illustrates an example where requirements A and B replace each other partly. The same requirement A therefore has a different utility relative to the reference systems S and $S_B$. The utility estimated by $u_A(S_B) = U(S_{AB}) – U(S_B)$ differs from the dissatisfaction $d_A(S_A) = U(S_A) – U(S) = u_A(S)$. This shows that it is important to clearly define the reference system against which a requirement´s utility is estimated because it influences the output. Several authors do so in their work, for instance defining the reference system as one made up of all “mandatory requirements” [3].

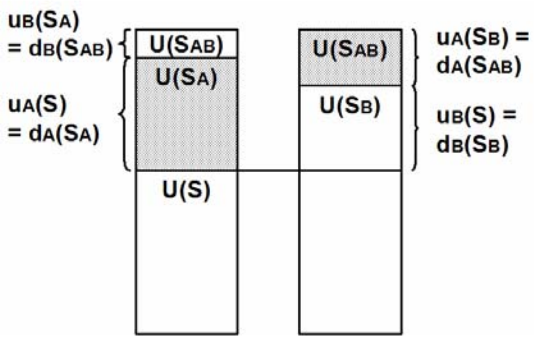


**Fig. 1**: utility estimation of requirement A relative to two different reference systems

Sometimes, a requirement´s utility is assumed to be its fixed characteristic, and utilities are summed to calculate the utility of a set of requirements [2,3,9]. But if in Fig. 1, we added $U(S_B) + u_A(S)$, we would not get $U(S_{AB})$. Such an approach neglects dependencies among requirements, which must be considered [10,11].

The consequence of this complexity is that in practical work, the stakeholders are confronted with having to estimate utilities relative to a complete system draft, and probably they will have to do repeat it for different reference systems. There are several simplifications for reducing this complexity such as grouping requirements (e.g. in features or super-requirements [10,11]), estimating relative priorities [1] or concrete values like low/ average/ high, or intervals [10].

For describing cost and other prioritization criteria, we expect that a function similar to the utility function must be assumed.

1. Karlsson, J., Ryan, K.: A Cost-Value Approach for Prioritizing Requirements. IEEE Software 14 (5) (1997) 67–74
2. Kazman, R., Asundi, J., Klein, M.: Quantifying the Cost and Benefits of Architectural Decisions. In: Proc. Int. Conf. Software Eng. (2001) 297-306
3. Ruhe, G., Eberlein, A., Pfahl, D.: Trade-Off Analysis For Requirements Selection. Int. J. of Software Eng. and Knowledge Eng. 13 (4) (2003) 345-366
4. Wohlin, C., Aurum, A.: What is important when deciding to include a software requirement in a project or release? In: Proc. Int. Symposium on Empirical Software Eng. (2005) 237 – 246
5. http://en.wikipedia.org/wiki/Utility_function (april 2006)
6. http://en.wikipedia.org/wiki/Constant_relative_risk_aversion (april 2006)
7. Robertson, S., Robertson, J.: Mastering the Requirements process. Addison-Wesley (1999)
8. Rupp, C: Requirements-Engineering und –Management - Professionelle, iterative Anforderungsanalyse für die Praxis. Carl Hanser Verlag München Wien (2004)
9. Xie, N., et al: SQUARE Project: Cost/Benefit Analysis Framework for Information Security Improvement Projects in Small Companies. Technical Note CMU/SEI-2004-TN-045 (Software Engineering Institute, Carnegie Mellon University, 2004)
10. Davis, A.M.: The Art of Requirements Triage. IEEE Computer 36 (3) (2003) 42 – 49
11. Dahlstedt, A.G., Persson, A.: Requirements Interdependencies - Moulding the State of Research into a Research Agenda. In: Proc. Int. Workshop on Requirements Engineering: Foundation for Software Quality REFSQ (2003) 71-80.