What if "business process" is the wrong metaphor? Exploring the potential of Value Based Requirements Engineering for clinical software

Thomas Wetter[a] and Barbara Paech[b]

[a]*Institute of Medical Biometry and Medical Informatics, Heidelberg University, Heidelberg, Germany*
[b] *Institute of Computer Science, Heidelberg University, Heidelberg, Germany*

Abstract and Objective

Low satisfaction of clinical software users suggests to challenge the prevailing "business process" metaphor underlying most present clinical software. We outline why Value Based Requirements Engineering, a new direction in software engineering, has the potential for better accepted software and how an empirical approach can lay the foundations for exploring that direction. Relations between value inventories and software properties are target of the present project phase.

Keywords:

Clinical software, Value based requirements engineering

Introduction

Early clinical software systems were the results of straightforward programming. When systematic development methods came into use the otherwise approved business process metaphor was also applied in medicine. Respective models assume that an organization enacts processes and assigns roles to employees. They take the physician captive of an overwhelming machinery to whose functioning s/he has no choice but to contribute a transaction here and there. Professional organizations such as IMIA and its Workgroup "Organizational and Social Issues" are aware of the problem, but try to solve it by supportive measures ***within*** the process or workflow metaphor. Since the problem has persisted for decades it seems overly conservative to believe that the processes metaphor has only been applied inappropriately. It rather seems timely to contrast it with a totally different one.

Approach

Value Based Software Engineering has emerged as an alternative. Its native form starts with business values of organisations which is, however, as top down as business process orientation. We, therefore, take it one step further: a bottom up approach starting from physicians' personal values. This is not an article about having applied value based requirements engineering. It rather wants to make the case that it is possible to try and that it is worth trying.

"Value" has a distinct meaning substantiated in various investigations in the social sciences. "Fundamental values" denote very basic orientations such as power, achievement, benevolence (10 in total). There are well approved marker sentences to distinguish the 10 fundamental values. High positive ratings for specific subsets of the marker sentences coincide with fundamental values prevalent in individuals.

In concrete situations fundamental values transform into modes of behavior such as honesty, helpfulness etc. Values give rise to behaviors and to attitudes towards objects and situations in work and other environments. They form a calibration backbone for emotions, attitudes, and actions. Modes of behavior can be probed to interpret behavioral or emotional reactions to work situations, among them software usage situations. Concretely, an individual that is diagnosed as benevolent in the fundamental values questionnaires is likely to affirm behavior patterns of the helpfulness segment and will react positively to work or other situations where s/he can support individuals in need and to software features that support him or her to provide help. An individual who is diagnosed as power driven will affirm and reject other situations and patterns. Values have been shown to be a rather stable facet of personality and at least somewhat coherent within individuals, groups, and societies. Questionnaires for probing individuals as to their value inventories are available.

Limited studies have investigated value inventories of physicians. They deliver role stereotypes such as *good gamaritarian, professional artist,* etc (anaesthetists sample).

A variety of more or less standardized and calibrated methods will be tried and explored as to their practicality for routine use and their appropriateness to be related to desired or not desired software properties. Such methods will be combined with observations and video recordings of usage of software, where utterances and gestures of (dis-)content shall be used as markers for preferred or rejected software properties. Together they provide orientation for the creative work and major target of the project: establishing relations between values and software properties. Different software types (decision support, discharge letter, ..) will be compared and may reveal that values help more with some and less with other software types.