Social Issues in Requirements Engineering[1]

Joseph A. Goguen
Department of Computer Science and Engineering
University of California at San Diego
La Jolla CA 92093-0114

Abstract

This note is intended to stimulate awareness and discussion of social issues in Requirements Engineering. It classifies issues according to what groups are involved, and sketches some issues in each class.

1 Introduction

A computer-based system is built for people and by people. It is intended to serve some human purpose, and it exists in some human context, whether it is a university information system, jet fighter, heart pacemaker, or stock arbitrage system.

The goal of Requirements Engineering is to determine what properties a system should have in order to succeed. Sometimes this is relatively straightforward; but often, it is not. Some of the most vexing difficulties are social, political and/or cultural. Many requirements engineers feel that such issues fall outside the scope of their profession, and instead fall under management, interpersonal skills, or ethical quibbles; some even argue that social issues *should* be ignored in the requirements process, in order to make it as clean and technical as possible.

A different view is that social issues are *inherent* to the requirements process, because the needs that drive that process are necessarily embedded in the social, cultural and political world of those who want the system, and hope to benefit once it is built.

A further complication is that the requirements process itself is social, because it minimumally involves interaction between engineers and clients, and it often involves extensive interactions among many others.

These considerations suggest that Requirements Engineering can never be an entirely formal process, because its very nature is to discover client needs, and reconcile them with technical possibilities. An alternative view, often implicit in works on formal methods, is that the requirements process is merely a matter of making *a priori* properties explicit.

To a first approximation, three major groups participate in the requirements process: the client organisation; the requirements team; and the development team. Of course, each of these may have subgroups, they may overlap, and there may be other interested parties (e.g., regulatory bodies), but this three fold division gives rise to six areas where social issues can arise: within the client organisation; within the requirements team; between the client and the requirements team; between the development and requirements teams; within the development team; and between the development team and the client. Some issues in each category are briefly evoked in the following five sections; the last two are treated together. The final section is a short discussion of how we might deal with social issues in Requirements Engineering.

2 Issues within the Client Organisation

Given a large organisation with rival departments and a complex hierarchy, it may not be obvious who or what to consider "the client". A further difficulty is that "users" may be completely different people from those interacting with the requirements team. It is important to identify the chain of command, who the proposed system is supposed to benefit, and who will actually use it. The workers who will use the system may need to be brought into the requirements process; in some cases, their exclusion can lead to their subverting the system, and causing failure. Other difficulties can arise if there are no staff currently in place with experience relevant to operating the new system.

The requirements process often reveals or highlights difficulties within the client organisation, and in such cases, it may be important to the success of the project for the requirements team to facilitate communication. JAD (or RAD) workshops provide many cases in point. It may be crucial to ensure that such difficulties are not merely buried. It can be useful to explicitly discuss the value system of the client organisation, or even do some empirical work to discover what it really is. ([5] discusses some related issues and examples.)

The introduction of a new system may have profound effects on the client organisation; e.g., some formerly public information may now be private, or *vice versa*. The success of the system may depend on understanding the effects of these changes; of course, this can be very difficult. Possible approaches include iterative design, e.g., gradually introducing change through prototypes.

3 Requirements Team Issues

The requirements team itself is a social organism, often part of some larger organisation. Relevant issues include how work is organised, what methods and notations are used, and more subtly, what beliefs team members have about organisations ([1] shows that theories of social organisation are implicit in requirements methods, and affect the requirements process).

4 Issues between the Client Organisation and the Requirements Team

Professional and commercial relationships underlie any requirements task. These entail financial arrangements, ethical obligations, legal safeguards, and personal relationshps; all these can be problems if things go wrong. Some more subtle issues include hidden agendas, the effects of change, and confidentiality.

Requirements engineers often encounter situations where some aspects of their work are supposed to be hidden from certain people. It is not usually problematic to keep commercially sensitive data from competitors. But ethical issues are raised if, for example, workers are not to be told that the new system will eliminate their jobs. Ethical issues also arise if the requirements team concludes that a system is infeasible, or severely flawed, while the manager to whom they report insists that this is not an acceptable finding; his motivation might be to save face, to discredit someone else, or even to harm his company. It is also common for people in the client organisation to hide certain information (e.g., see [2]), and it is not uncommon for requirements engineers to be denied access to certain individuals or groups. More subtly, members of social groups often behave differently when outsiders are present.

Another common problem is dealing with changes in the client's view of what is needed. These often occur as a natural result of increased understanding generated by the requirements process; they can also arise from changes in the client organisation or its social context (e.g., tax laws, or its competition). Ideally, change would be considered inevitable, and often desirable; but unfortunately, the work statement for requirements efforts often does not allow for change, and the requirements process is often unable to accomodate change easily.

5 Issues between the Requirements and Development Teams

System developers can often benefit from direct contact with the requirements team, to discover the reasons behind choices, to resolve inconsistencies, to get more details, etc. Unfortunately, many system development processes do not allow on-going communication of this kind. Indeed, it is common for a huge document to be delivered to the development team, after the requirements team has already been disbanded. Sometimes even access to individual members of the former requirements team is forbidden. This is an example of the excess optimism of which the waterfall model is an extreme case.

6 Development Team Issues

As with any group, social issues can influence the quality and quantify of work. If the team is divided or demoralised, there may be a high turnover, and deadlines may slip. Documentation can be a problem, as developers often dislike doing it.

In many large projects, much of what developers actually do could be classified as Requirements Engineering. For example, requirments may have to be reconsidered in light of a deeper understanding of the project's possibilities and limitations, or the client may have new ideas. Often this work is done without invoking combersome official procedures. If direct contact between the client and the developers is not allowed, many difficulties can arise. Similarly, much of so called maintenance may actually fall within the concerns of Requirements Engineering.

7 How can we Handle Social Issues?

Traditional Requirements Engineering methods tend to ignore social issues. More recent methods may use techniques inspired by psychology or sociology, including interview, workshop and questionaire techniques, and human information processing models. We have argued [3, 4] that such techniques have systematic limitations, and that in some cases, they should be supplemented or replaced by techniques based on ethnomethodology, ethnography and sociolinguistics.

Traditional sociology is "normative," that is, it presupposes a fixed set of categories, such as status, gender, and class, and then interprets phenomena in terms of these "factors." By contrast, ethnomethodology attempts to determine what concepts, and what methods of legitimation, are used by the members of groups themselves. This can lead to a very different, and more accurate, requirements process. A potential problem is that organisations often have their own normative concepts ("myths"), which do not reflect what actually happens, but rather reflect members' beliefs about what should happen. An important research problem is how to integrate approaches based on ethnomethodology with more conventional requirements methods and notations.

References

[1] Matthew Bickerton and Jawed Siddiqi. The classification of requirements engineering methods. In Stephen Fickas and Anthony Finkelstein, editors, *Requirements Engineering '93*, pages 182–186. IEEE, 1993. Also, Technical Report, Centre for Requirements and Foundations, Oxford University Computing Lab, 1992.

[2] Richard Boland. In search of management accounting: Explorations of self and organization. Technical report, Case Western University, 1991.

[3] Joseph Goguen and Charlotte Linde. Techniques for requirements elicitation. In Stephen Fickas and Anthony Finkelstein, editors, *Requirements Engineering '93*, pages 152–164. IEEE, 1993. Also Technical Report, Centre for Requirements and Foundations, Oxford University Computing Lab, 1992.

[4] Paul Luff, Marina Jirotka, Christian Heath, and David Greatbatch. Tasks and social interaction: the relevance of naturalistic analyses of conduct for requirements engineering. In Stephen Fickas and Anthony Finkelstein, editors, *Requirements Engineering '93*, pages 187–190. IEEE, 1993. Also Technical Report, Centre for Requirements and Foundations, Oxford University Computing Lab, 1992.

[5] Enid Mumford. *Designing Human Systems for New Technology.* Manchester Business School, 1983.