Gender and Software Engineering

Edeltraud Hanappi-Egger
Vienna University of Economics and Business Administration, Austria

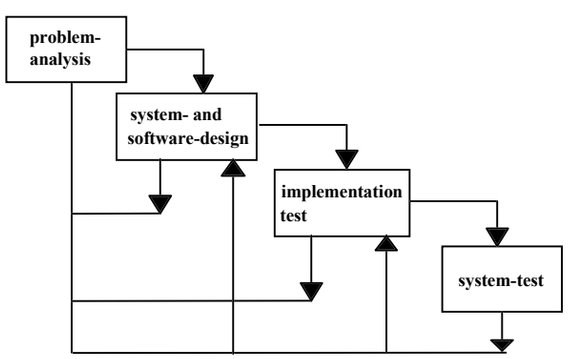
HISTORICAL BACKGROUND

The history of software engineering is not very long, but rather intensive. In the beginning of the 40s, programming was seen mainly as manual, electronic engineering job, done by women, in particular mathematicians (compare e.g., Schinzel and Zimmer, 1998). Changing more and more to an intellectual, mathematical, high status task programming started to be male-coded and remained to be done by men.

The first computer systems were developed for military purposes. The end products seemed to be not adequate to the expectations of the users. Most system engineers did not care about the users' needs, nor did they consider any kind of documenting their own thoughts and works as being necessary. This led to the so-called "software-crisis" in the sixties and raised the discussion of adequate–in terms of more structured and more transparent—methods in software engineering. One important result of this discourse was the classical "waterfall model" (see Figure 1).

Software development is seen as sequence of single phases. The output of the preceding phase delivers the input to the proceeding phase, returning to a former state is in principle possible, but there is also a point of no return caused by cost constraints. Generally, this approach considers system development as step-by-step procedure without iteration, only small punctual adaption should be done. User-involvement is often limited to the first phase, namely to the problem analysis. In order to define the functions of the computer systems the users have to serve as information source, describing the current working process, and formulating system requirements. The underlying engineering perspective clearly makes it to the designers' task to come up with a technological solution. The contact with users is limited to the first step when specifying the functionality of the system and the to the last step when the system is delivered.

Figure 1. The waterfall model

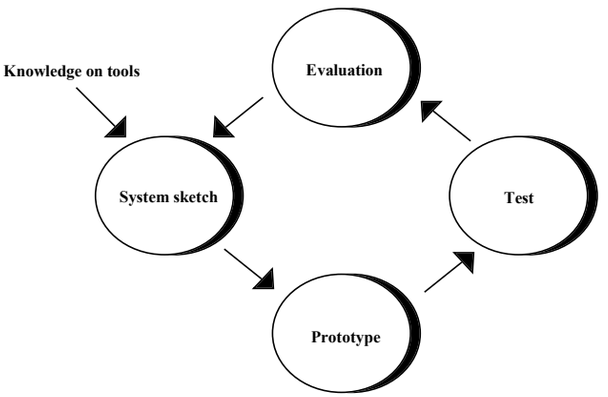


This treatment of system design and development still lead to inadequate products and as a consequence more attention to the users' needs was necessary. In particular, in the growing commercial and non-military application fields it became clear that there must be a stronger cooperation of engineers and people from the concerned organizations.

The new "socio-technical system approach" stresses the relation of human aspects and computer aspects trying to find an optimal "fit" for both (Mumford, 1971). Contrary to the purely "engineering" perspective, it has become accepted that social dynamics play a crucial role in the design and development procedure. Consequently new methods of system design integrating social aspects have been developed. In particular, the communication problem between users and engineers was identified as main source for badly specified systems. Thus, the involvement of the users during the whole system design—and development process became a necessity—this was the birth of (e.g., the idea of participatory design) (Floyd & Keil-Slawik, 1983) and of methods called "prototyping" (see Figure 2.

In the literature, there are some different models (e.g., incremental model, spiral model, etc.), which do have the same basic ideas of the prototyping approach: the iterative involvement of the future users of a system. A system sketch, a main kernel is implemented in cooperation with the users. After a

Figure 2. The prototyping approach



testing phase with users (which is supervised by system-experts or tutors), the evaluation of the first prototype will lead to a step-by-step extension of the system.

Besides communication features this approach requires a new understanding of system design: Computer scientists are not the experts from the application field. Consequently, they have to cooperate with people actually working in the according environments.

In the last decades, this approach of considering software engineering as organizational modeling was further developed. It became clear that system development is organizational development (compare e.g., Briefs, Ciborra, & Schneider, 1983; Kling, 1984). The design and introduction of a computer system imply changes in the organizational structure and organizational aspects strongly determine the design and specification of technological systems.

Summarized it can be stated that in the last decades the view on system development changed a lot. Users were more and more involved—ranging from pure observation at the beginning of structured system design to contemporary user-centered design methods (see also Kuala, 2003). In particular, the view that computer system design is organizational design implies many decision-making competencies of the users. Thus, it could be stated that there is a new understanding of constructing technological artifacts leaving the path of "objective" modeling following the view of "negotiated constructions." The development context in terms of who is involved, which competences do these people have, which budget and time constraints play a role and the like influences the shape and function of systems. Consequently the process of system design is embedded in social settings and therefore always very specific (see also Hanappi-Egger, 1998, 2000). The author demonstrated this with computer science students: All students got the same task of programming software systems. At the end of the semester, there were as many (different) solutions as development teams.

The next question immediately is if system design is the result of social processes, which role plays "gender?" In other words, if the participants of system design are "gendered," how does this influence the design and development of software systems?

THE ROLE OF GENDER IN SOFTWARE ENGINEERING

From an organizational point of view, there are specific gender patterns and gender relations structuring organizations and consequently the social interactions. In other words, "sex" is associated with social constructions of gender roles and behavioral expectations. Biologically identified "women" are more likely expected to be feminine and men are expected to be masculine. This social mechanism tries to match the biological and the social gender of human beings. At the same time there is a gender hierarchy subordinating "female" coded activities and capabilities. West and Zimmerman (1998) call this "doing gender" referring to the fact that there are processes in organizations, taking place producing and reproducing gender relations (see also Hanappi-Egger, 2003).

Combining the two approaches, namely that gender plays a crucial role in social settings in general and that system development is a social activity (as presented in the historical overview) leads to the statement that gender has to play a role in the design and development of technological artifacts, too (for a general discussion see Hanappi-Egger, 2003).

In order to elaborate this statement, an analogy to the built environment can be drawn: From a design perspective in the built environment Kennedy (1981, p. 76) exemplifies gender-specifics in architectural practice: The female principles consists of more