IT Design for Amateur Communities

Cristian Bogdan
KTH, NADA, CID/IPLab
cristi@nada.kth.se

Doctoral thesis abstract
Draft, tentative date for defense: September 2002

IT design for communities has emerged lately as an important direction of research in Human Computer Interaction and Computer-Supported Cooperative Work. Addressing computer support for physically co-located communities (community networks), communities of practitioners inside large professional organizations, or geographically distributed communities defined by common interests (network communities), the new topic is hard to delimit but is nevertheless a clear and vigorous line of research.

This thesis focuses its interest on socio-technical aspects of work in geographically distributed communities of amateurs. Observing the incipient stage of the CSCW community research agenda, one of the goals is to contribute to the agenda through field study of amateur communities. Studying amateur communities is bringing to focus communities that do not emerge from the net , and have strong off-line , in real life aspects to their activities. This comes to fill a gap between the real-life workplace, usual in the CSCW literature, and the virtual community. It also calls for widening the research focus from computer-mediated communication (CMC) software, which is employed by almost all communities discussed in the literature. As the online aspect fades in importance, so does the identification of the community with a certain software system. Another contribution to the research agenda is proposed by emphasizing the theme of self-sustainability of software design and development practices in the community, after the researchers leave the setting.

Amateur cooperative work is examined by ethnographically-inspired field study in the amateur radio community and three geographically dispersed student communities. Although not computer-mediated, the amateur radio community raises interesting research questions due to its emphasis on communication, the central role of technology, its remarkable endurance over long periods of time and its world-wide area of dispersal. The study of amateur radio lead to the development of a set of concepts intended to help in IT design. These concepts are then refined throughout the thesis.

The amateur radio work is seen as very *resource-efficient*, with one activity (e.g. communication) intrinsically adding value to others (e.g. the technology of communication). This inspires us to conclude that, as resource-scarce settings, amateur settings are likely to benefit from design strategies that would support such mutually-implicative activities.

Another finding is the importance of *contingency* in the amateur activity. Contingencies are valued and welcomed by members and their negotiation fuels future work, the work of addressing *challenges*. In amateur radio, challenges (e.g. the achievement of long-distance radio connections with low emission power) are inexhaustible (or hardly exhaustible), yet actionable. In the aforementioned mutually-implicative manner, challenges lead to the intrinsic durability of the community, and to the natural formation of subcommunities. This brings us to design strategies such as identifying the core challenges of an amateur community, designing for their preservation and development. As known from other community studies, members are in a continuous *learning* process, and at every learning stage, new kinds of interesting *contingencies* have to be negotiated. The importance of *pioneering* when identifying new challenges is also emphasized: amateurs enjoy having a *research value* for large sections of their community and (if possible) for the society at large.

The study of student organization work sees the members as mainly amateur arrangers of large projects aiming at international student exchange. Studying their work confirms and improves the design concepts that were drawn from the amateur radio study, yet they provide a new perspective, of conflicting challenges between groups and subcommunities. Also, challenges are of a more heterogeneous nature than those in amateur radio. Such conflicts and heterogeneities should be carefully observed and valued by designers.

Long before the existence of research interests in computer support for communities, many amateur communities (including radio amateurs) have built and shaped software to support their activities. In order to better support amateur communities with software, the historical evolutions of software are examined in the context of the three student organizations. One finding is the importance of amateur developers, their active role in making and sustaining design decisions and their difference in *challenge* from the rest of the members. In many instances, members refrained from asking for more features or for bug fixing, knowing that the software is developed and maintained by fellow volunteers. Another important finding is that, once a procedure is supported by a system, the system serves also as a handbook for *learning* that procedure, gradually replacing the actual book of rules and regulations in the members practice. Thus *learning* takes place through *engagement* with the software, in addition to learning from peers.

After observing the amateur settings through field study, the researcher became a participant designer in one of the student communities, working with the introduction of participatory design practices in the community, as well as with the design of technologies. Participatory design is recommended and pursued by many researchers of community. The experience in the student organization found important a-priori problems for participatory design. Although the strong democracy suggested a fertile ground for such practices, although many problems reported by cooperative design practitioners in industrial settings are not present (e.g. the difficulty to form design groups), problems were found in other areas. Many such problems are related to learning: members had to learn both the practices of project arrangement and the practices of design. This was even more pronounced for developers, who also had to learn about the development tools. Other problems came from the member s needs to see higher-resolution prototypes i.e. to work with more concrete objects than the paper mock-ups.

This experience, together with the field study observations, contributed to the design of tools for prototyping and tailoring software in the organization. In the *resource-efficient* manner learned from amateur radio, software is prototyped and design is continued in use by employing the same tools, therefore sparing learning time for developers and users who turn to development. Tools were designed to be usable right away by as many students from the setting as possible, supporting *learning* also at later stages. The tools form an open-ended generic system, providing a variety of *challenges* to the member developers, also providing them with opportunities to *pioneer* new areas and consider themselves as having a *research value* for their community, and (due to the generic and open source character of the software) the world at large. All systems of the organization were ported by the members to the new technology and are in production at the moment. The set of tools has been formally evaluated through questionnaires, with promising early results. Another promising result is the increase of the number of developers and the involvement of designers in lower-difficulty development tasks. Since self-sustainability is a long-term concept, it is too early to evaluate if the self-sustainability objectives of design have been met.

Designing for self-sustainability of design and development practices inspired from the self-sustainability of practices in the amateur radio and student communities has thus revealed a set of core design concepts for design in amateur communities (resource efficiency based on mutually-implicative activity, challenge and contingency, pioneering and research) that form the main contribution of this thesis. A strategy of designing by challenging the users (challenging design) has been developed and evaluated in the design exercises. The results support the idea of designing for communities that are not immersed in a certain CMC system (as most communities studied by current research are). Furthermore, many of the needs for design identified have been for single-user systems, leading to the idea that one of the ways of supporting the community is to actually design for the individual member. The thesis also proposes the research of historical evolutions of community software as a method that can inform community software design.