

Some PD Advice

Gro Bjerknæs
UNIVERSITY OF SALFORD
SALFORD, ENGLAND

PD is different from conventional design approaches in that users and system experts share the responsibility for the quality of the design proposal and the implemented system. Thus both system experts and users get new roles in the system development process. The system experts cannot make final design decisions on their own. This is a difficult position because it requires a great deal of trust in the users. The users, on the other hand, are responsible for the system they get. If they don't like it, they cannot blame the system experts.

In this way, PD brings users and system experts together in mutual commitment. The users must learn about technology from the system experts in order to understand what computer technology can do for them, and the system experts have to learn about the application domain from the users in order to build a flexible and efficient system that fits the users' needs. If a PD process is successful, it will contribute to a new perspective both on one's own work as a system expert, but also on the diversified and complex work of the users.

However, even if the users have commitments to the project, they are also responsible for doing their jobs. And even though system experts share the responsibility with the users for the design of a computer system, the system experts will nevertheless be responsible for its implementation—often based on deadlines scheduled beforehand. Thus, both users and system experts have commitments that compete with the mutual commitment of the PD project. This tension can lead to frustrations and even conflicts. Therefore, it is important that PD projects also design the environments of the projects to cope with these tensions.

Here I provide some advice on how to avoid a failing PD project based on experiences collected during the four-year Florence project.¹

What to take care of when the project is established:

- *Have management support.* If the users are committed to both the project and their colleagues, in terms of doing their work, their colleagues will not have to do their job. If the project has management support, it is not easy for the

users to use work load as an excuse for not having kept their commitments.

- *Specify in a contract how much time the users can/shall spend on the project.* In addition to rights and obligations for the participating parties, a contract should specify the amount of time the users can/shall use on the project per week. Otherwise, the users will give priority to their ordinary work tasks, making it difficult for them to do project tasks between meetings. When the effort is quantified, it is more legitimate for the users to do project tasks. While there are some enthusiastic users who do the work in their leisure time, it is difficult to have participation on equal terms without a specific contract.

- *Have a steering group in which conflicts can be discussed.* Sometimes the tension experienced by the different groups can lead to frustration and conflicts. In that case it is important to have a contract and a forum at which the problems can be discussed. The steering group must consist of people who have the power and resources to intervene if necessary.

- *Be sure the required equipment is available for system experts and users.* This may seem obvious, but it is not always so. If the users are supposed to test the system (or parts of it, or early prototypes) as part of the development effort, the users must have the equipment to do that.

What to take care of during the project:

Listen to the users. Take the users seriously! If they say there is something they don't want, don't do it, even if it seems to be the perfect computer system for their needs from a system expert point of view. If a system is built that the users don't want, the users are overruled, and it is no longer a PD project.

On the other hand, do not do everything the users propose. The users are not always aware of the consequences of their proposals, and they can often be conservative in that they do not utilize the possibilities given by computer technology. A systems expert should contribute knowledge about technological possibilities and limitations, and general knowledge about what kind of computer systems are useful in different settings. If the system expert just uncritically does what the users want, it is not a PD project.

Don't forget the users. Even if PD means mutual commitments and shared responsibility, users and system experts

will have different tasks in the project, and the work load will not be equally distributed. The period when the system is implemented is a critical one. The users have put much effort into the design of a system, and then it seems as if nothing happens. Thus, if the system cannot be implemented in an incremental way, so the users can see progress, the system experts have to communicate and inform the users about the progress. Otherwise the users will feel left out, not viewing the system as their system anymore, and it will no longer be a PD project. This can be critical for the project as a whole. After all, the most important part of the project is when a system is made usable, and that is the time the users will do most of the work. If they do not feel committed anymore, the result can be a system no one wants to use.

If this advice is followed, some traps can be avoided that would otherwise turn a PD project into a conventional design project with user participation.

¹The Florence project was initiated in 1983 by the Information systems group at the University of Oslo's Department of Informatics and was sponsored by the Royal Norwegian Council for Scientific and Industrial Research. The project concluded in 1987.