3.6 User Involvement

Continued and effective user involvement throughout the project is an important factor in maximizing the chance of success. The traditional waterfall lifecycle is less amenable to user involvement during the whole project and hence is less likely to stay in line with the changing user requirements. A prototyping approach is normally dependent upon continuing user involvement and by its very nature encourages it. However, care has to be taken to ensure that the users have sufficient time to perform their roles effectively. In prototyping, the evaluation of the prototype requires significant time from the users.

Users can be involved in projects at various levels and can play many different roles.

Some approaches to software development (e.g. DSDM) directly involve users in the development team, so that they then have a significant opportunity to influence the way the project proceeds, perhaps by identifying difficulties and suggesting more acceptable alternatives.

It is important that users who have such a significant influence on the direction of the project should understand the organizational requirements and the needs of fellow users. Direct involvement of users is more likely to be successful if they are considered as full members of the project team and if they are genuinely empowered to represent the organization and make decisions within clearly defined operating parameters. There is always a danger that users who become members of a project team cease over time to represent the user perspective effectively, as they begin to view themselves more as a team member and less as

a user representative. One way of overcoming this tendency is to rotate membership of the development team among a group of users. However, this can result in a loss of continuity.

A more satisfactory approach is to scope each activity so that a user team member can see it through to completion in a reasonably short time, say within three months.

User involvement as participants only in fact gathering is at the other end of the spectrum. Such users may provide information about current working practice but they have little or no influence on the design of the new system. In this situation users are likely to be concerned about the project and may fear the effect it will have on their jobs.

As a result, they may be less co-operative and will be less willing to take ownership of the new system when it is installed.

Even when users are not invited to join the project team, effective participation can still be encouraged through a consultative approach. Procedures are set up so that users are able to review certain aspects of the systems development, provide feedback, give their views of the system and then are able to see the response of the project team to their feedback. In some circumstances a large part of the task of requirements capture may be delegated to users. They are likely to feel a strong affinity to the delivered system, provided that it satisfies the requirements they specified.

Whatever form of involvement users have with the project, it is important that their role is carefully explained and that training is given as required. Some large organizations have gone to the trouble of training users so that they understand the terminology and models used by the systems developers. Users must also be given the time to participate.

It is no good expecting users to review requirements documents effectively if the only time they have available is during their lunch break.

Users who participate in a systems development project can be selected in various ways. They can be designated by management as being the most appropriate representatives or they can be selected by their peers. In either circumstance they must be genuine representatives of the user perspective.

A responsibility assignment matrix (RAM) is a useful way of delineating roles and responsibilities for a project. The matrix normally has tasks or deliverables listed in the left-hand column and roles listed along the top row. Roles should be distinguished from individuals in the project team. A particular role can be performed by several people and one individual may occupy several roles. One form of RAM is a RACI matrix. RACI is an acronym for responsible, accountable, consulted and informed. A simple example of a RACI matrix is shown in the table below. The responsibilities that are normally assigned in the matrix are listed below with examples from the table below.

- **Responsible** – the role that does the work to perform the task or produce the deliverable. For example, the applications development role builds the software increments.
- **Accountable** – the role that is accountable for the completion of the task or production of the deliverable; only one accountable role is assigned to each task or deliverable.
- For example, the project sponsor is accountable for the Project Initiation Document and must ensure that it is produced by the project manager.
- **Consulted** – the role whose opinions are sought. For example, the user representative is consulted about the Use Case Model.
- **Informed** – the role that is kept up to date on progress. For example, the user representative is informed about the development of the software increments.

Deliverable/Role	Project Sponsor	Project Manager	Applications Development	Analyst	User Representative	Development Manager
Project Initiation Document	A	R	I	C	C	C
Project Plan	C	A	C	C	C	C
Use Case Model	C	A	I	R	C	C
Priority Requirement List	A	I	I	R	C	C
Software Increments	I	I	R	C	I	A

Key: R – Responsible
A – Accountable
C – Consulted
I – Informed