Chapter 4

Specific Techniques 2 : Group session approaches

Objectives

- To address one cause of process failure:
  ..that there is a failure to realise that appropriate human communication mechanisms need to established as part of the requirements process. If different interest groups do not communicate effectively with each other, each will seek to exert power and influence over the others...Gasson, 1995, Markus & Bjorn Anderson 1987
- To present a case study which illustrates the problem of inappropriate human communication mechanisms
- To highlight the role of group session approaches in assisting with this problem
- To discuss the role of the facilitator in group sessions
- To introduce the key features of Joint Application Design (JAD) sessions and their role in requirements
- To illustrate the role of Quality Function Deployment (QFD) in requirements
- To give an overview of Cooperative Requirements Capture (CRC)
- To present some practical techniques for Cooperative Requirements Capture

4.1 Introduction

This chapter begins with a case study of a multidisciplinary research and development project. An analysis of the progress of the project is presented. The analysis shows that although the project began well with agreement on the process model to be adopted, the two main groups drifted apart. Two different requirements documents were produced, one from each group, neither group developed any real understanding of the needs of the other group, and both groups attempted to take control of the development process. This resulted in eventual failure and abandonment of the project. The analysis explains how and why this happened. The main problem is identified as a lack of appropriate human communication mechanisms within the project.

The main body of this chapter is dedicated to three different approaches to requirements in which appropriate human communication mechanisms are provided. The three are placed together under the heading of group session approaches because each is centred around holding a series of group meetings in which the role of facilitator is explicitly defined. The approaches are introduced here because they contribute to requirements twenty one to twenty seven from the 'wish list', that is, human communication techniques which:

21. support construction of appropriate requirements teams
22. support identification of stakeholders
23. support the development of a 'shared meaning' of the system being specified
24. encourage intuition, imagination and common sense among participants
25. support communication between people from a diversity of backgrounds

26. support facilitated meetings with predefined agendas and problem solving strategies

27. support the development of listening skills among participants

The three approaches were chosen for the following reasons:

- JAD (Joint Application Design) is a well established group session approach in which the role of group members and the facilitator are well defined. In addition JAD is an example of a structured analysis approach to requirements.
- QFD (Quality Function Deployment) is a group session approach which is gaining recognition within software development circles. The role of the facilitator is well defined. In addition QFD is an example of a quality approach to requirements.
- CRC (Cooperative Requirements Capture) is a group session approach in which the role of group members and the facilitator are well defined. In addition CRC is an example of a Human Factors or HCI approach to requirements.

4.2 An illustrative problem situation

The purpose of this section is to present an analysis of a particular project in which two different interest groups were involved. Although the project began with good intentions on both sides, the lack of appropriate communication mechanisms resulted in each group developing a different view of the requirements. This in turn led to attempts to exert power and influence over each other through producing different requirements specifications and following parallel development processes. The two groups never achieved a common understanding of the requirements and although two prototypes were developed, the project was eventually abandoned.

This analysis is taken from the work of Susan Gasson, Gasson 1995, selected extracts are reproduced here with permission.

4.2.1 A study of a multi-disciplinary research and development project (Gasson, 1995)

This study was concerned with a multi-disciplinary research and development project, based at a UK University, to design a computer-based system to support interactive student learning. It was conducted through an analysis of the design documents produced by the project team, and through a series of interviews with project team members. As the interviews took place after a decision had been made to abandon the project, some of the team-members' *post-hoc* attitudes could be interpreted as defensive: triangulation[1] was used between interviews, to derive a representative picture of the project. Although an external project sponsor was involved, the sponsor's involvement was limited to contact via progress meetings - for this reason (and as access was complicated by the sponsor's withdrawal from the project), the sponsor's contact-staff were not interviewed as part of this study.

[1] Triangulation is a process of verifying the results from more than one perspective.

From the beginning of the project, there was an explicit recognition of the need for a high degree of user-involvement, to permit evaluation of the student-learning benefits of the system. A decision was made by the project manager, a senior academic psychologist, to recruit equal numbers of psychologists and IS professionals onto the team and to use an iterative prototyping model for the system development process: this model is given in figure 2. It is clear that, from the beginning, the psychologists were seen as proxy (and powerful) users by both themselves and the IS professionals on the team: they were there both to evaluate the learning benefits of the target system and to ensure that the system was designed for optimum usability.

4.2.2 The research framework (Gasson, 1995)

The research framework used for analysis in this study was that proposed by Markus & Bjorn-Andersen (1987) and shown in figure 1. The influence of users in development decisions is constrained by information systems (IS) professionals who may exert power over users in four ways:

- technical power may be exerted in advocating a particular course of action without providing users with the evidence to make their own evaluations
- structural power may be exerted by developing IT policies and practices which constrain user choices
- conceptual power may be exerted by shaping users' concepts of what IT can provide
- symbolic power may be exerted by shaping user values with respect to IT (normally through the provision of system exemplars).

		Target of power exercise	
		Issues of fact	Issues of values
Context of power exercise	Specific development project	Technical	Conceptual
	IS management policy	Structural	Symbolic

figure Types of Power Exercise (Markus & Bjorn-Andersen, 1987)

Figure 1

4.2.3 Research Findings (Gasson, 1995)

A comparison of the intended process-model (figure 2) with the actual process-model (figure 3) of the project is illuminating. From the beginning, there appears to have been a dichotomy of approach between the two disciplines, despite attempts by the project manager to co-ordinate process-paths, which reflected team-members' disparate interests. Two separate system requirements documents were produced, one reflecting innovations in the *use* of the system, another reflecting its basis in leading-edge *technology*. Even when the results of the initial requirements documents were combined, two rival requirements specification documents were produced, each reflecting only part of the other perspective

It would appear that team-members from neither discipline fully understood the requirements of the other discipline and both sub-groups attempted to resolve the resulting cognitive dissonance by prioritising their own requirements. The need for IS professionals and users to learn from each other during system design and development is a common thread in information systems literature: Eason (1982) highlights the time-lag between developer understanding of technical potential and user understanding, while Curtis et. al. (1988) discuss the critical role of the “expert designer” - who has prior experience of a particular application-domain - in educating other, technical team-members. However, this team lacked the integrative mechanisms necessary for such learning. Both disciplines attempted to control the development process: the psychologists by agreeing project task-structures and deadlines with the project-sponsor, the IS professionals by using the problematic nature of the unproven technology to separate the technical development processes from learning-evaluation.

Figure 2 The proposed process model for the R & D project, Gasson, 1995

The integrated design and development processes from the intended process-model (figure 2) became split into two, separate process-loops, controlled by the two, separate halves of the project-team in the actual process-model (figure 3). In response to the psychologists' attempt to exert structural power (by defining project tasks), the IS professionals gained control of the process by using structural dependencies between the tasks. The technical nature of the production of prototypes for evaluation gave the IS professionals the ability to exert technical power, as the psychologists did not have the expertise to produce these prototypes. Although there was a concerted effort, on the part of the psychologists, to participate in the design of the initial prototype ($Prototype_1$ in figure 3), this appears to have been thwarted by their dependence upon the IS professionals to configure the technology.

There appears to have been an implicit agreement between the two IS professionals working on this stage of the project that the first prototype was not intended to be incorporated into the target design, but was produced as a diversionary tactic, to occupy the psychologists while the IS professionals proceeded with the 'real' design. While this was partly a negative reaction, on the part of the IS professionals to what were perceived as unrealistic deadlines for the initial prototype (which had been set by the psychologists, in their attempt to gain control over the project), the IS professionals frequently used the term 'flower arrangers' to refer to the psychologists on the team - a revealing metaphor for their perception of the relative value of the contribution of technical and user requirements. When asked explicitly why the design and lessons learned from the first prototype were not used for the second prototype, the response from one of the IS professionals was:

> "Well the cycle broke down because it was such a naff prototype. I think we just generally ignored any requirements that came out [*from the psychologists*], because we had much better ideas that we felt were ready to go: what we wanted to do for the first 'real' prototype. Obviously our minus one [*$Prototype_1$*] was produced - but we generally just disregarded it".

Figure 3 The actual process model for the R & D project (Gasson, 1995)

The use of the name “minus one” for Prototype$_1$ reveals its perceived lack of relevance for the intended system outcome: at the same time as the psychologists were evaluating this prototype, the IS professionals were engaged upon the development and evaluation of a prototype for a completely different system design (Prototype$_2$ in figure 3). This was not communicated to the psychologists. Thus the IS professionals were able to exert symbolic power, by shaping psychologists' expectations of the system: the psychologists were more likely to accept design suggestions from the IS professionals following evaluation of the first prototype, as anything had to be better than the existing design!

The psychologists attempted to exert conceptual power over the IS professionals by the performance of field studies on commercially-available systems for similar purposes. However, these were not read by the IS professionals, who exerted their own conceptual power by prioritising technical requirements over user-requirements when selecting appropriate technology. They were able to do this as the psychologists had been placed in a weak position structurally: the evaluation report from the first prototype was not completed, as it became clear at this point that the design of the first prototype had been abandoned. The evaluation results (and by association, the psychologists' contribution to the project so far) were therefore meaningless. Once a second system prototype had been produced, technical and usability evaluation still took place as two disparate processes, conducted in isolation from each other, with neither process informing the other. It would appear that the project was then abandoned, for a number of complex reasons - not least

that communications between team members had almost completely broken down. Although the ostensible reason given was the withdrawal of the project sponsor, the project failed to find another sponsor because the body of work produced proved insufficiently coherent to attract further funding.

One of the most striking issues which arose from an analysis of the interviews was the formality of communication between the two ‘sides’ of the project team: the psychologists and the IS professionals. The team size was relatively small - at no time did the core team exceed six members - yet most of the communication between IS professionals and psychologists appears to have been via formal specification documents. It appeared that the IS professionals treated the psychologists on the team as proxy users and therefore dismissed them as an unnecessary distraction from the core task of designing the technical system, while the psychologists felt frustrated and resentful at their dependence upon the technical expertise of the IS professionals: both sides tried to legitimise their priorities by the production of ‘official’ project documents - an explicit attempt to exert conceptual power. In pursuing reasons for the integrative failure of the two disciplines, it was observed that the team members had been accommodated in two separate offices: one for the psychologists and one for the IS professionals. When asked why the disciplines had not been mixed in their accommodation, the internal project manager commented that the two disciplines had refused to share an office, but the psychologists' perception was that the IS professionals had refused to share an office as they did not want to ‘waste’ time in educating the psychologists in the technology to be used. The psychologists made repeated requests for tuition in the technology, but these were refused on the basis that meeting them would divert the IS professionals from the tasks necessary to meet project deadlines - another exercise of structural power.

This case study revealingly illustrates the exercise of all four types of power on the part of the IS professionals, most of which appeared to be explicitly directed to exclude the proxy users who were perceived as an unnecessary diversion from the ‘real’ processes of design. The psychologists were observed to have attempted to exert both conceptual and structural power over the IS professionals, but were unable to exert technical or symbolic power, as they were in a position of dependency with respect to the technical expertise which was required to exert these two types of power in a context where the design process was defined as primarily technical. Even when the psychologists had exerted structural power by taking control of the project deadlines, the IS professionals were able to subvert this control by the use of their technical power, in producing a throw-away prototype.

Also revealed is the importance of recognising both IS developers' and users' learning processes as legitimate design activities. Both interest groups on this project appeared to act as they did because they had insufficient understanding of the requirements and the necessary activities of the other group. If the project process-model had included tasks designed to educate each of the two groups in the others' domain, it is likely that the project would have been more successful. However, integrative communication mechanisms and organisational structures were also needed, to allow the two groups to reach a common set of interests. As the IS professionals were permitted to work together,

as a separate organisational unit, and as system design was perceived by the whole project team as being primarily a technical activity, the IS professionals were able to re-define and control the central processes of the project and to exclude the evaluation and use studies which had formed the *raison d'être* of the research project. A comparison of the intended process model with the actual process model (figures 2 and 3) illustrates to what extent the IS professionals were able to exclude users from the central processes of design and decision-making.

4.2.4 Conclusions from the illustrative problem

A number of points can be drawn from this case study in terms of the themes of this book. Firstly that a process failure can occur even though a project may start with an agreed process model. Secondly that a process model is useless unless the human communication mechanisms are put in place to support the project team in following the process. Thirdly that although the two groups had a shared objective in the development of the proposed system they did not share a vision of what the system should do and why.

The next two sections introduce some of the characteristics of group session approaches and details of the role of the facilitator. In essence the process model will provide the agenda for the group session or more likely, for a series of agendas for a series of group sessions. Each group session would normally be preceded by some preparatory work and would be followed by some further work. The role of the facilitator is seen as crucial to enabling effective communication between different interest groups.

4.3 Characteristics of Group Session Approaches

There is an increasing recognition of the importance of group session approaches to requirements capture and analysis. Group session approaches to requirements normally involve a number of people coming together in a meeting situation. As with any group meeting there needs to be an agenda and someone to act a chairperson or facilitator of the meeting. For the group session to be successful people must listen effectively, feel free to express ideas and must be empowered to participate in the process and the outcome of the meeting. The session will be judged to be successful if there is an agreed and documented outcome.

In this section an attempt is made to identify a number of general characteristics of group sessions.

Sharing of workspace

Group sessions usually take place in rooms where the furniture can be arranged to facilitate face to face communication between group members. Often the tables and chairs are set in a horse-shoe shape so that all members can see all other members and see the facilitator. The lighting, heating and general ambience of the room must allow for intense and uniterrupted periods of discussion.

Communication between group members

Group session meetings provide a rich medium of communication. A great deal can be achieved in a setting in which people can see each other and can be sensitive to each others behaviour and reactions. Someone seen tapping their fingers on the table may be clearly annoyed, someone yawning may have lost interest in the discussion, another person leaning forward and pointing may be agitated or trying to make a point forcefully.

Sharing of information
Sharing of information is essential to prevent unnecessary duplication of effort and to ensure that all members can access the same information. The group needs facilities to support the documentation of, navigation and retrieval of that information. This is often acheived through the use of whiteboards and flipcharts. For example, results of brainstorming sessions are often documented of sheets of flip chart paper and subsequently pinned up around the room for group members to view and to retrieve the information as required.

Coordination and control of shared objects
In a group session there must be only one version of the object under discussion. For example, consider a requirements team attempting to develop a task hierarchy diagram, a number of versions of the diagram can accumulate and become difficult to manage; more than one person may be modifying the diagram at the same time and there may be a proliferation of associated notes, papers and diagrams which become difficult to maintain.

Decision making
Central to the requirements group is the ability of the group to reach a decision. The decision may be concerned with the objectives of the common task of the group, the method of working to be adopted by the group and the choice of the group members. Specific decision making techniques may be employed, such as, social judgement analysis, the Delphi technique and the Nominal Group technique, see Viller 1991b for further details.

Organisation and common understanding of the work process
In group session approaches the method or technique used largely determines the agenda. It general the group will need to agree on the role of each individual, set specific objectives and deadlines, and decide upon some way of keeping informed as to how each person, and the group, is progressing. The facilitator plays a key role in this.

Facilitation
The facilitator plays a key role in helping the group to reach decisions, in managing the shared workspace by deciding who will write on the whiteboard or other shared workspace and in facilitating communication between team members. However, it is not only the trained facilitator who facilitates in group sessions, members of the team themselves may choose to facilitate ‘from the floor’. The role of the facilitator is discussed in more detail in the next section.

4.4 The Role of the Facilitator

This section draws largely on work by Viller (1991a) and Viller (1991b).
A feature common to many group session approaches to requirements engineering is the notion of someone who's role is to assist the process of group working, generically referred to as a facilitator. The term facilitator itself denotes a set of skills and behaviours that may be applied by a group-worker, teacher, manager, therapist, coordinator, and so on. The application of these skills may be different in the various contexts. Nevertheless, "facilitator" is a readily identifiable, common 'core' of skills and behaviours that may be used by any of the above.

The Shorter Oxford English Dictionary defines facilitate as "To render easier; to promote, help forward". The role of a group facilitator, therefore, is concerned with assisting the other group members in performing their collective task as a group.

At the initial stages in a group's lifecycle, the relationship towards the facilitator may be all that is common to the other group members, and thus the facilitator becomes the group's central person. A facilitator, with their knowledge of group process, can utilise this position to improve group cohesion, and for the setting of group norms. As a group develops, individuals will identify themselves more as members of the group, and the common relationship of everyone towards the central person will become less important (Douglas, 1970). During these middle stages of the group's lifecycle, the central person's role is much more that of enabler, sitting back from the group and only intervening when necessary. Finally, as the group nears its end, the role of the central person becomes more important again, as he or she assists the other members through the process of winding-up the group. The precise role played by the facilitator at this stage will depend upon the circumstances in which the group is breaking up; for example, whether or not the group has fully achieved its purpose (Douglas, 1970).

Opinions differ on the facilitator's status within a group. Some of this difference can be explained by the 'bias' of the source. For example, if the facilitator is to perform some leadership function for the group - as in management situations - then he or she will be in a position of power over the other group members. Conversely, if he or she is someone who is brought in from outside of the group as a professional facilitator, then his or her function will be more of an assistant to the group, helping the other group members to achieve their objectives without having any stake in the outcome. This second example describes the facilitator's role in its generic sense, the key factor being that the facilitator is concerned with enabling the *process* of the group achieving its aims, whilst having no stake in the *content* of these aims.

Whilst the dynamic aspect of group work is one of its advantages, problems can develop, and when they occur the facilitator's role takes on greater importance. It is necessary for any facilitator to be able to recognise when a problem is developing, and to also have the skill and knowledge of how to enable the group to deal with it.

Any action that a facilitator takes to 'correct' group process problems is known as an *intervention*. Five *Generic Problem Syndromes* (along with their symptoms, possible causes, and possible interventions) have been identified by Westley & Waters (1988) - presented in table I

'Multi-Headed Beast' syndrome

SYMPTOMS	Digressions; interruptions; multiple topics; no listening; no integration of ideas.
POSSIBLE CAUSES	No agreement on agenda; no process design; mixing problem-solving strategies.
POSSIBLE INTERVENTIONS	• Suggest round robin to clarify task • List perceptions of task • Seek synthesis (rephrase, find continuities, categories) • Formulate/reformulate agenda

'Feuding Factions' syndrome

SYMPTOMS	Repetitious arguments; open attacks, anger.
POSSIBLE CAUSES	Hidden agendas/power struggles; fear of change.
POSSIBLE INTERVENTIONS	• Stop action: "we're having difficulty agreeing on a solution..." • Allow individual to privately list criteria • List criteria independently of alternatives • Measure alternatives against criteria.

'Dominant Species' syndrome

SYMPTOMS	'Plops'; 'unequal air-time'; passive/aggressive body language; withdrawal
POSSIBLE CAUSES	Dominance: not heard, frustrated Withdrawn: afraid, frustrated, insulated
POSSIBLE INTERVENTIONS	**Direct:** question/poll under-participators; thank/limit over-participators **Interpretative:** At end of meeting, share perceptions on levels of participation • self rating • round robin on views • solicit norms on participation

'Recycling' syndrome

SYMPTOMS	'Broken record' behaviour; irritation with lack of progress; failure to gain consensus.
POSSIBLE CAUSES	Ideas not being recorded; confusion about problem-solving process.
POSSIBLE INTERVENTIONS	• Introduce/reintroduce problem-solving steps • identify which issues belong to which steps • identify 'where we are, where we've been, where we're going'.

'Sleeping Meeting' syndrome

SYMPTOMS	Long silences; absence of energy/ideas; withdrawal.
POSSIBLE CAUSES	Fear of volatile issue; hostility; depression, fatigue.
POSSIBLE INTERVENTIONS	• Describe observation - 'blocked condition of meeting' • Suggest mood-check • Then: - take a break - address underlying problem - decide on action plan to rectify • and/or - return to task, allotting time to address the problem at end of meeting.

Table I: Generic Problem Syndromes, after Viller, 1991b

A facilitator will usually have at his or her disposal a number of techniques for assisting a group in the decision making process. These may vary from simple “brainstorming”, to more complex *structured group process* methods, see Viller 1991b for further details.

The requirements engineering techniques which follow all provide some level of guidance on the role of the facilitator. The group session approaches described below are JAD, QFD and CRC.

4.5 Joint Application Design (JAD)

JAD was introduced briefly in section 1.7.4 as a structured analysis approach. IBM's Joint Application Design (JAD) (August, 1991) draws users and information systems professionals together to jointly design systems in facilitated group sessions. Gibson et al., 1987 claimed that JAD studies report 20 to 60 percent increases in productivity over traditional design methods. Further, that JAD promotes cooperation, understanding, and teamwork among the various user groups and information systems staff. JAD defines six different roles which should be represented at a group session, these are: the session leader, the user representative, a specialist, an analyst, an information systems representative and an executive sponsor. JAD teams are given guidance and proformas which can be used as a basis for the agenda for group sessions, however, teams are encouraged to customise these according their problem situation.

The JAD facilitator is referred to as a session leader. According to Crawford, 1994 the session leader manages the process, facilitates debate and preparation of documents. Within the group session the role of the facilitator is similar to that described in section 4.4, however, Crawford describes a role which also includes the facilitator taking explicit responsibility for activities outside the meeting situation. The session leader is expected to liaise with the JAD sponsor, to reach agreement as to who should attend meetings, to be responsible for agreeing the agenda with participants, to agree on allocation of work to participants between meetings and to ensure that all appropriate documents and presentations are prepared on schedule. Crawford, 1994, contains advice on how to run a JAD session and how to become an effective session leader. Crawford also provides a number sample workbook pages for specific applications of JAD.

The JAD approach can be used at various levels of detail, starting with the business vision and concept analysis through to requirements analysis and specification of designs. JAD activities would be needed for each level of detail. August, 1991, describes typical JAD activities as JAD/Plan followed by two JAD /Designs. Each activity is composed of three phases: customization, session and wrap-up.

A JAD/Plan session usually lasts from one to five days, depending on the size and complexity of the system. The session leader guides participants through eight tasks (August, 1991):

- conduct orientation

- define high-level requirements (including objectives, anticipated benefits, strategic and future considerations, assumption and constraints, security, audit and control requirements)
- bound system scope (including business flow diagram, system users and locations, out-of scope functional areas)
- identify and estimate JAD/Designs
- identify JAD/Design participants
- schedule JAD/Designs
- document issues and considerations
- conclude session

The JAD/Design session usually lasts between three and ten days. The session leader guides participants through the following tasks (August, 1991):

- conduct orientation
- review and refine JAD/Plan requirements and scope
- develop workflow diagram
- develop workflow description
- identify system data groups and functions
- specify processing requirements
- document issues and considerations
- conclude session phase

The JAD sessions are accompanied by workbooks which contain a collection of proformas for the teams to complete either during the session or as part of the follow-up activity. Examples of proformas include participant matrix forms, issues forms, estimating assumptions forms, screen layout forms, report layout forms, interface description forms, function description forms.

In addition to providing support for a group session approach to requirements JAD also contributes to the techniques 'wish list' in the following way:
For the process of requirements it provides:

- support articulation of the product concept
- support problem analysis
- support feasibility studies and cost-benefit analyses of options
- support analysis and modelling
- support documentation of requirements
- support a systematic step by step approach
- standardised ways of describing workproducts
- procedures for maintaining workproducts

In terms of human communication it provides:

- support for identification of various viewpoints

- support for reconciliation of viewpoints
- supports the user in reviewing models developed
- support users in analysing their own problems and identifying the need for change

JAD helps to develop knowledge of:
- Relevant structures on the users' present work
- Visions and design proposals
- Overviews of technological options

4.6 Quality Function Deployment (QFD)

QFD was introduced in section 1.7.8 as a quality approach to requirements. Quality Function Deployment (QFD), (Sullivan, 1986), originated in the Japanese car industry as a means of translating customer requirements into appropriate technical requirements throughout the development and production of a product. QFD is based on group sessions in which the 'House of Quality' (see figure 5) is used as a focus of attention. The House of Quality is centred around a matrix which shows the relationship between the customer requirements and the proposed product features.

In QFD the customer requirements provide a central theme and are used as a basis for setting targets for the design and implementation. Traditional QFD is split up into four iterative phases, namely product planning, parts deployment, process and control planning, and production planning (Thackeray and van Treek, 1990). Basically all four phases are similar, each has it's own House of Quality and associated group sessions. The people who attend the group session will be those who are responsible for that particular phase of the product and who need to come to an agreement concerning their actions within that phase.

Betts, 1989, encourages people to apply QFD to software projects. Figure 4 illustrates how she sees QFD fitting into the software development lifecycle.

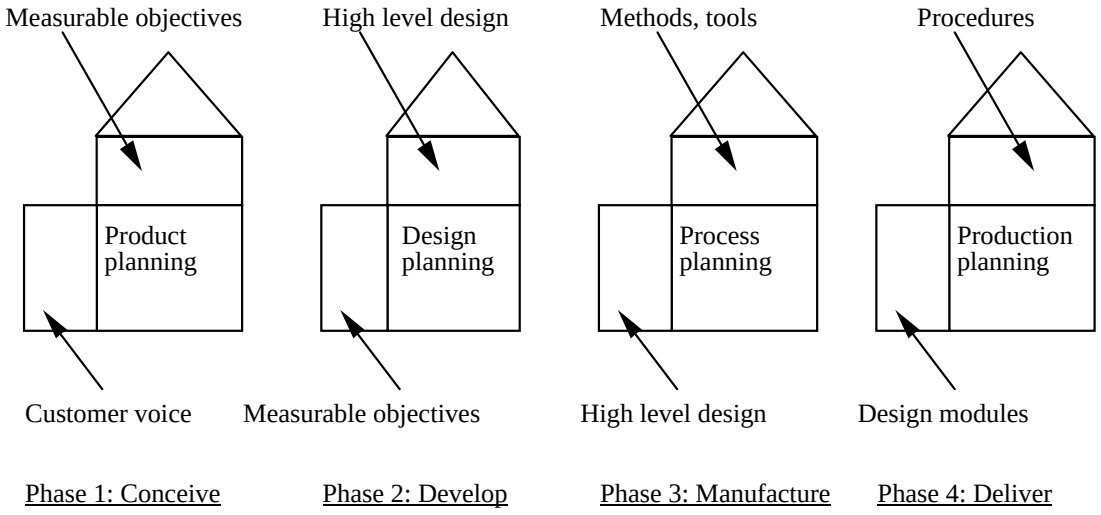

Figure 4.QFD and the Software Engineering Lifecycle, after Betts, 1989

The customer voice provides the driving force for identifying the measurable objectives for the product, the objectives are then used to drive the high level design, and so on, such that the customer voice acts as a driver for the whole software development process.

Zultner, 1989, provides more detailed guidance on how to use QFD on a software project, he illustrates how to incorporate data flow diagrams and entity relation diagrams into the design phase. In Zultner, 1993, he demonstrates how a wide range of tools and techniques which can be linked to QFD in order to develop a Total Quality Management approach to software development. A key theme of the paper is that of process improvement through project teams identifying and setting their own targets while at the same time being aligned to the organisation's vision for improvement.

Reports of the use of QFD in various parts of the computer industry (Daetz, 1989, Cohen, 1988) have claimed reductions of 17% in product definition time, and clear traceability of requirements from initial design through to full production.

Marsh, 1991, describes facilitation of the team effort as critical for successful completion of a QFD study. Proper facilitation, he claims, will help the QFD team fully integrate the talents, skills, and creative potential of each team member. The facilitator has a coordinator's role in the planning, design, execution and completion of a QFD project. The facilitator is described as a neutral, non-evaluative, and non-manipulative group focal point. Marsh , 1991, has produced an excellent guide called 'Facilitating and training in QFD' which gives details of the roles and responsibilities of both the facilitator and the project manager of a QFD project.

The remainder of this section describes the activities associated with phase 1, the planning phase. The outcome of the planning phase is the overall customer requirement Planning Matrix, see figure 5. This translates the voice of the customer into counterpart control characteristics; that is, it provides a way of turning general customer requirements (drawn from market evaluations, comparison with competition, and market plans) into final product control characteristics. (Sullivan, 1986)

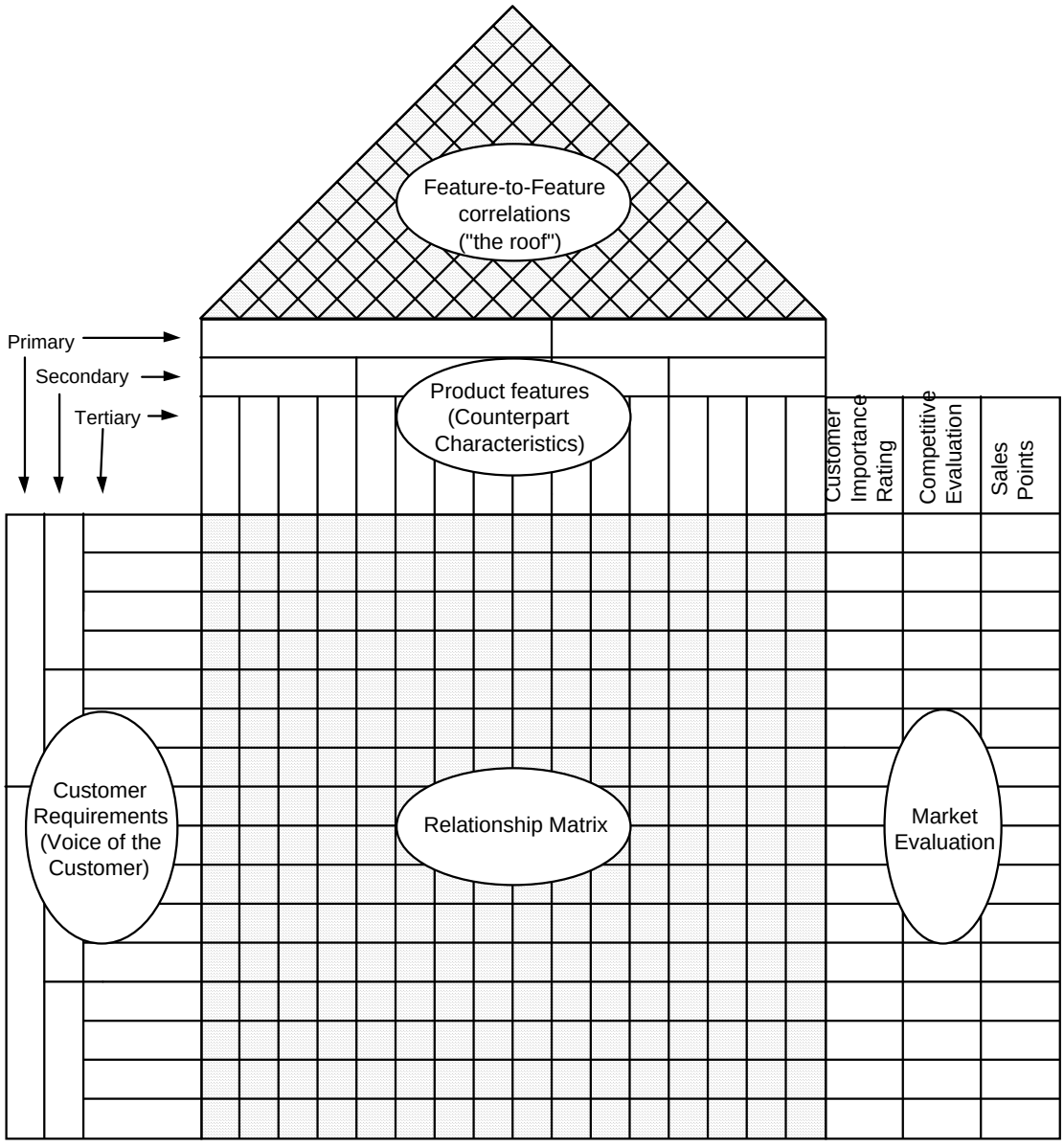

figure 5 The House of Quality for the Planning Phase

Putting the planning matrix together involves eight steps:

Step 1: The first step is to state the product requirements in customer terms. These are entered into the left hand column of the matrix under the heading ‘customer requirements’ in figure 5. The requirements may be grouped together into primary , secondary and tertiary requirements. Primary requirements often being quite general, such as ‘easy to learn’, the secondary and tertiary requirements would be more specific.

The information needed for step one usually comes from a variety of sources, no specific guidance is given as part of the method, despite the fact that this is obviously a critical step in the whole process.

Step 2: Step two is to list the ‘product features’ across the top row of the matrix. These are the final product characteristics which will needed to meet the stated customer requirements. The characterisics must be capable of being expressed in measurable terms.

Step 3: Step three is to develop the ‘relationship matrix’. This shows the relationship between the ‘customer requirements’ and the product features (the final product control characteristics). The relationship is expressed as being strong, medium or weak, figure 6. The benefit of filling out this relationship matrix using appropriate symbols (Sullivan, 1986) is that it quickly whether the final product control characteristics adequately cover the customer requirements. Absence of symbols (or a majority of ‘weak relationship’ signs) indicates that some customer requirements are not addressed. At this point, product characterisics may need to be modified or supplemented to ensure that all customer requirements are adequately addressed.

Step 3 - Develop relationship matrix between customer requirements and final product control characteristics

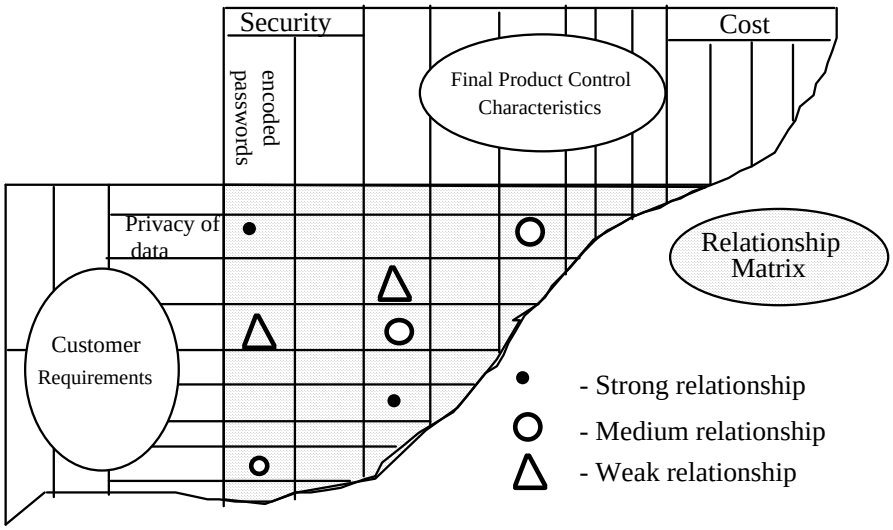

figure 6 Step 3 in the planning phase

Step 4: Step four is to add the market evaluation. This shows the customer rating of the importance of each customer requirement and an evauation of how well the competition is doing in meeting each requirement. See figure 7.

Step 4 - Enter market evaluations

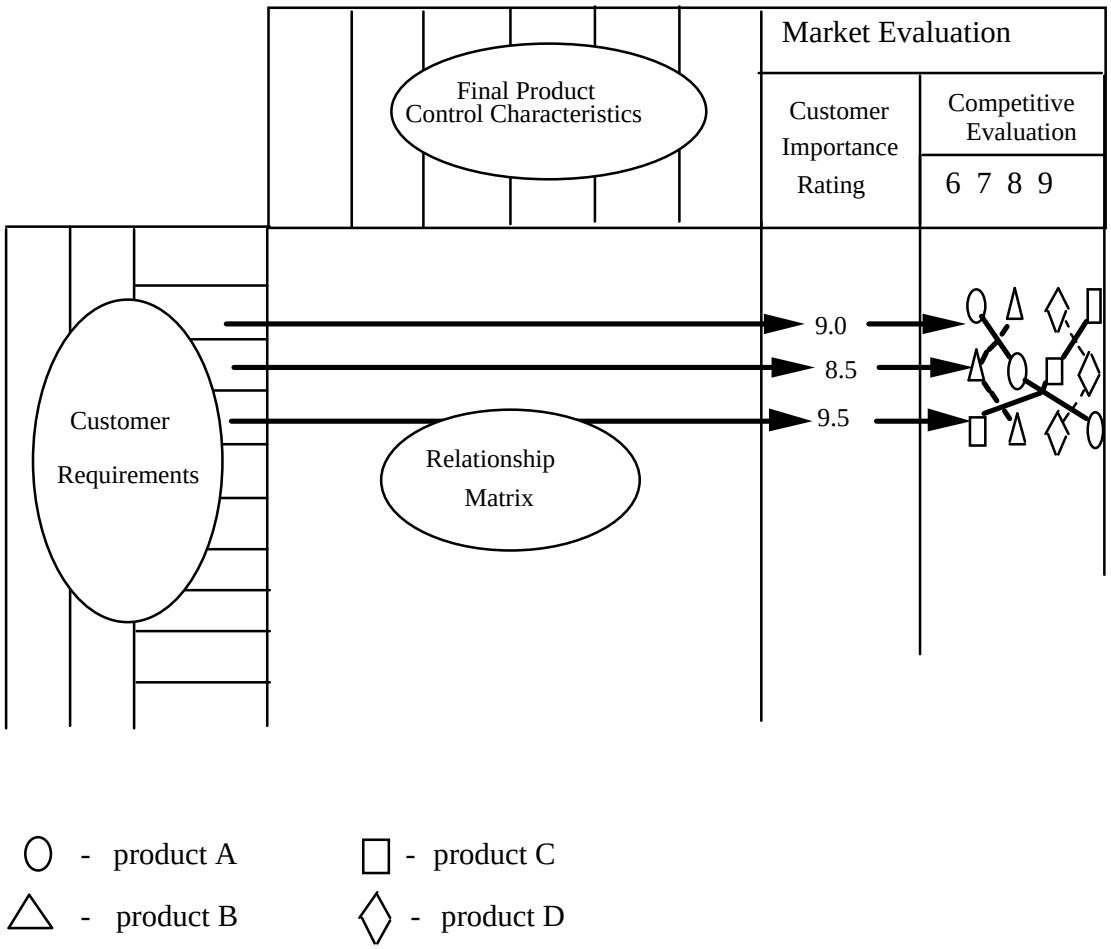

Figure 7 Step 4 of the Planning phase

Step 5: Step five is a comparison of the final product control charactistics against the performance of the competition. This step involves identifying the numerical, measurable performance rating, for example, of each of product A, B, C and D against each of the product characteristics. This gives a measure of the level of achievement of the competition.

Step 6: Step six is concerned with using the analysis from step five and the customer importance rating from step four in order to identify the selling points for the proposed product. Statements such as ‘‘best-in-class’ at meeting customer requirement X’ or ‘lowest energy consumption components for meeting customer requirement Y’.

Step 7: Step seven now requires the QFD team to identify the actual measurable targets (control characteristic targets) which must be achieved. These targets are based on the agreed selling points, the customer importance rating, and current product strengths and weaknesses.

Step 8: Step eight in the development of the Planning Matrix involves the selection of product control characteristics that are to be deployed through the remainder of the QFD process.

In addition to providing a group session approach to requirements, QFD also contributes to the following requirements from the 'wish list':

- supports articulation of the product concept
- helps develop visions and design proposals
- supports identification of requirements for generic products
- is capable of working alongside market analysis techniques
- supports analysis of competitive products
- supports predictions of future users and future use, and estimations of future usage
- supports generic descriptions of typical users and groups of users
- supports identification and specification of quality attributes: usability, reliability, portability, performance, security, maintainability, acceptability and so on depending on the proposed system

Sources of additional information: For overviews of QFD see Sullivan, 1986, King, 1989. For guidance on how to conduct QFD see Bossert, 1991, Marsh, 1991, Zultner, 1989. For case studies see Cohen, 1988, Lecuyer, 1989. For discussion, announcements of conferences, workshops and publications see QFD-L list , 1995, also GOAL/QPC, 1989

4.7 Cooperative Requirements Capture (CRC)

The approach developed by the author through a number of collaborative projects is called Cooperative Requirements Capture (CRC) (Macaulay, 1993). This is a group session approach similar to JAD in that the role of participants and the role of the facilitator are clearly defined. These are not only users and designers but also include those with a stake in the system being proposed. A stakeholder is defined as all those who have a stake in the change being considered, those who stand to gain from it, and those who stand to lose. The CRC facilitator does not have stake in the proposed system , but does have knowledge of the method and the interpersonal skills required of a facilitator.

Some of the stakeholders identified have a direct responsibility for the design and development of the various system components and hence have a major interest in being involved in the requirements capture process. Others have a financial responsibility for the success of the computer system and therefore may also need to be involved. While others will be the recipients of the resulting computer system, they also have a major contribution to make in terms of specific task knowledge and the ability to assess the likely effects of the new system. The choice of representative stakeholders to attend the group sessions is of vital importance to the success of the session.

For example, in a requirements analysis for a proposed theatre box office system the following stakeholders were involved: From the theatre: the accountant, the box office manager, an experienced box office and the theatre manager; from the software house: the project manager, the designer, the senior designer and the user support person. Figure 8 illustrates the role of the CRC group session in bringing the two organisations together.

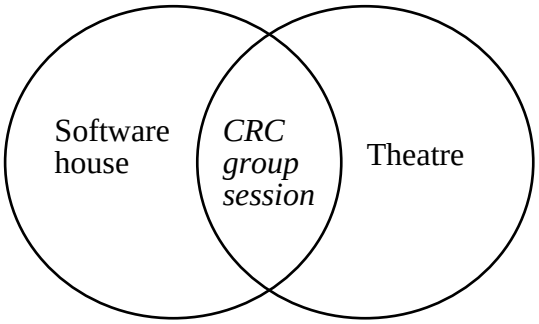

Figure 2: CRC group session: scenario 1
Figure 8

In contrast to this, Macaulay, 1994c reports a case study of the use of CRC to determine the information requirements of control room engineers in the year 2000. In this case three separate organisations were involved in the group session, one was the research and development unit associated with the electricity supply industry and the other two were separate electricity distribution companies. There were four types of stakeholders involved: (i) ‘strategic thinkers’ from all three organisations, these were people with a long term view of their company’s future direction; (ii) computer specialists from all three organisations, who would ultimately be responsible for the design and development of the proposed system; (iii) control room engineers from the two distribution companies who would be representative of the users of the proposed system and (iv) managers of the control rooms who would ultimately be responsible for the introduction of the proposed new system. The role of the CRC group session is highlighted in figure 9 below:

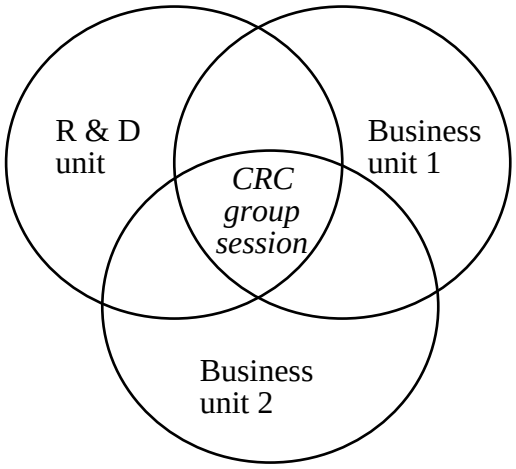

Figure 3: CRC group session: scenario 2
Figure 9

CRC is different from JAD and QFD in that the focus of attention is the user. The CRC method is as follows: (i) identify the problem; (ii) formulate the team; (iii) group session 1: explore the user environment; (iv) validate with users; (v) group session 2: identify the scope of the proposed system and (vi) validate with stakeholders. Each group session has a number of steps, for example, session 1 includes (i) the business case; (ii) workgroups; (iii) users; (iv) tasks; (v) objects; (vi) interactions and (vii) consolidation. Each step includes an introduction, brainstorming, prioritisation and generation of agreed descriptions using checklists and proformas which deal with user related issues. For a fuller description and a detailed case study of group session 1, see Macaulay, 1995.

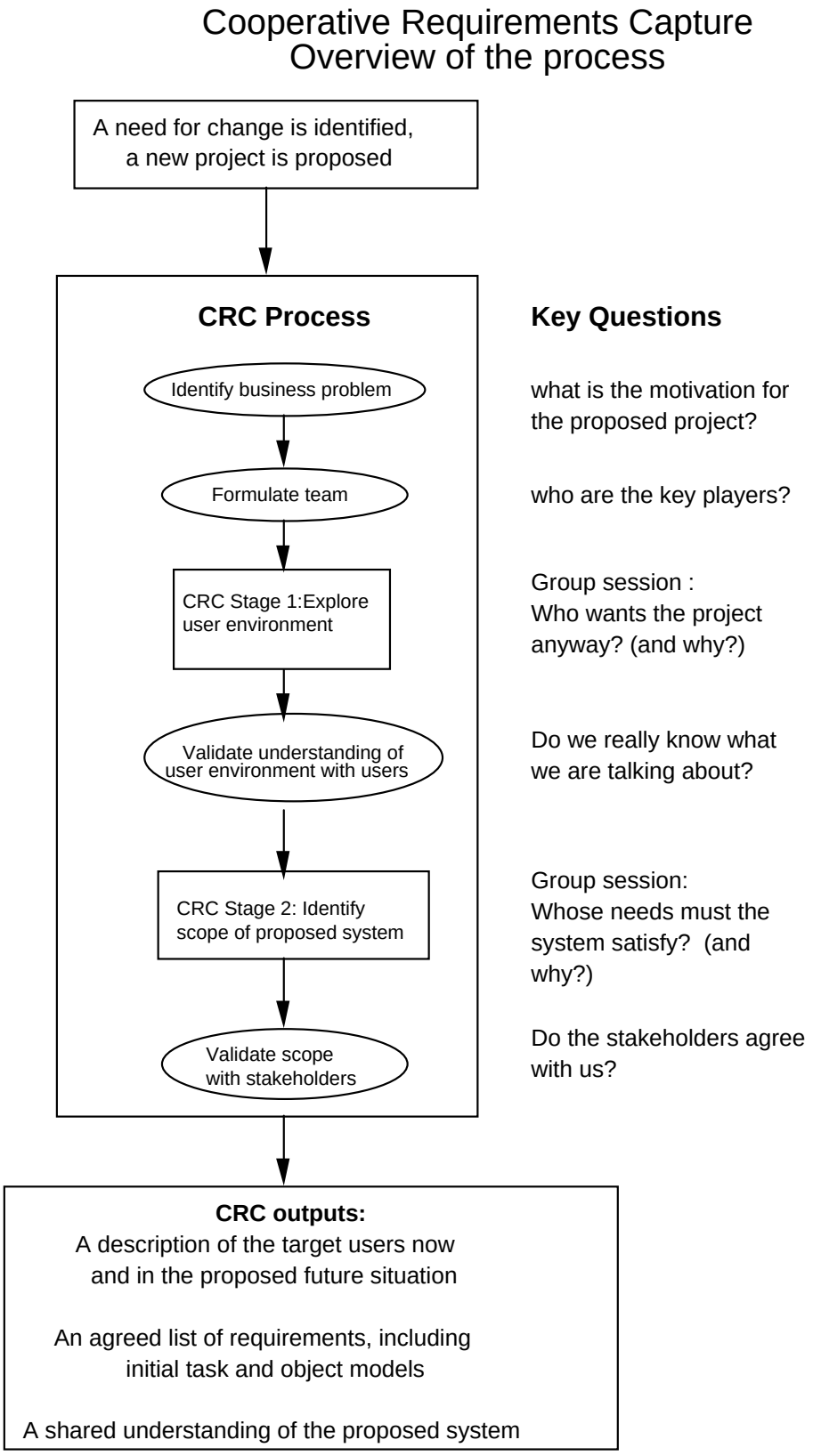
Cooperative Requirements Capture
Overview of the process
A need for change is identified,
a new project is proposed
CRC Process
Key Questions
Identify business problem
what is the motivation for the proposed project?
Formulate team
who are the key players?
CRC Stage 1:Explore user environment
Group session :
Who wants the project anyway? (and why?)
Validate understanding of user environment with users
Do we really know what we are talking about?
CRC Stage 2: Identify scope of proposed system
Group session:
Whose needs must the system satisfy? (and why?)
Validate scope with stakeholders
Do the stakeholders agree with us?
CRC outputs:
A description of the target users now and in the proposed future situation
An agreed list of requirements, including initial task and object models
A shared understanding of the proposed system

Figure 10

In the Cooperative Requirements Capture approach (CRC) the user and the user environment provide the focus of attention for the stakeholders, and helps them to develop a shared vision of the future system. They 'explore' the user environment together, they are encouraged to describe what users do now and to develop a vision of how things might change in the future. They develop a shared understanding of the potential for change and a shared terminology for discussing the problem domain. Figure 10 gives an overview of the CRC method.

The rectangular shaped boxes represent face-to-face meetings or workshops. Normally the meeting is supported by a trained facilitator, who guides the team through the main steps of the method and who also encourages all the stakeholders to participate. The oval shaped boxes represent activities which must take place either before or after a workshop. These activities usually involve some subset of the stakeholders consulting with others outside the immediate team. Communication and consultation is important in order to elaborate and validate the requirements before the detailed design begins.

Each part of the approach is described briefly below:

Identify Business Problem:
It is assumed that there will be some motivation for proposing that a future system be developed. That motivation may come from some specific business need, for example, the need to improve the speed of customer service at the check-in desk at an airport, or the need to provide bank customers with easy access to cash withdrawals. In some cases it may be that the commissioning organisation is planning ahead and that the business problem may refer to some future need, for example, electricity distribution companies may be asking: ‘What kind of computer support will control room engineers need in the year 2000?’. In other cases, it may be that only a small incremental improvement is needed, for example, an estate agent may already have a computer based system but may be wanting to improve the quality of the imformation stored.

If the commissioning organisation is a software house or computer company it may be that they are in possession of some innovative technology and that they wish to identify whether target users do have a need for that technology, and to be able to describe what that need is.

In all these cases, a business need is identified, whether it be an improvement in customer service, a future need, a small incremental improvement to the existing system or whether it be a need to use some technology which is available. There are many different motivations for proposing that a future system be developed. What is important to the application of the CRC approach is that the business need is articulated and described in such a way that it can be shared with the other team members.

Formulate Team:

This will normally involve the project manager or project initiator and the facilitator in identification of stakeholders and hence of the requirements capture team, ideally between six and nine stakeholder representatives will participate in the CRC process. The team would be drawn from each of the four categories of stakeholders identified earlier in the chapter.

CRC Stage 1: Explore User Environment

Exploring the users environment means that the requirements capture team must collectively investigate the organisational setting the target users are in and identify and describe what the target users do.The term 'explore' is used because the team is encouraged to 'find out' afresh, to share knowledge about users and to set aside preconceptions about what users need. They also assess the likely costs and benefits of change from the users point of view and produce a document recording the shared view of the users environment.

The structure of the 'User Document' is shown in table 2 below:

1. Management Summary
   (including the business case and a brief description of the proposed system)

2. Organisation/Workgroups
   - 2.1 Workgroup Control Sheets
   - 2.2 Organisation Chart
   - 2.3 Workgroup Table
   - 2.4 Workgroup Description Checklists

3. Generic Users
   - 3.1 Generic Users Control Sheets
   - 3.2 Generic Users Description Checklists

4. Tasks
   - 4.1 Task Control Sheets
   - 4.2 Task Hierarchy
   - 4.3 Task Description Checklists

5. Objects
   - 5.1 Objects Control Sheets
   - 5.2 Object Structures
   - 5.3 Object Description Checklists

6. Interactions
   - 6.1 User/Task/Object Interactions
   - 6.2 Initial List of Requirements and Attributes

7.Consolidation
   - 7.1 Statement of Credibility

7.2 Further Investigations Needed

8. Worth Proceeding?
    8.1 User/Stakeholder Perspective
    8.2 Business Perspective
    8.3 Plan of Action

9. Conclusion

table 2 List of Contents of the User Document

The User Document is produced by the Cooperative Requirements Capture team as a result of a two day face-to-face meeting.

Validate Understanding of User Environment with Users
Representative users do participate in the workshop but other users will need to be consulted or interviewed in order to ensure that the team has reliable information about all those users who may be affected by the system. After stage 1 of CRC, but before stage 2, validation of the information recorded in the 'User Document' should occur and it should be expanded or updated where necessary.

The techniques used for validation will depend on the specific problem, for a generic product further market research may be needed, for a bespoke system specific user interviews may be necessary. In any case, the extent of the information gathering task will depend on the extent of the knowledge and expertise of the stakeholders who took part in the workshop. A team who is highly conversant with their users may need to do very little validation. It is important to note that at this point in the development process highly detailed knowledge of all the users' tasks may not be necessary. The team needs to have enough, reliable information to be able to decide which users, which tasks and which objects need to be computer supported and to decide what the extent of that support should be. That is, they need enough information to decide the scope of the proposed system.

CRC Stage 2: Identify the Scope of the Proposed System
This also involves a two day face-to-face meeting of the requirements capture team and the use of proformas and checklists. At this stage the scope of the system is discussed. The scope of the proposed system is determined at a number of levels: Firstly the stakeholders decide which work roles are to be affected and then for each work role they decide what the role of the system should be in supporting that role. In particular the role of the system is decided in terms of the extent of task sharing and degree control and monitoring of tasks. This is discussed more fully in the next section 'Task Analysis and Allocation of Function'. The likely acceptability of this proposed change is considered. In addition for each work role identified an initial task model is produced, this helps to clarify and consolidate the understanding of the team with respect to specific roles. Secondly the team is asked to consider which objects from the user environment, that is those contained in the 'User Document', are likely to be of interest to the system, that is,

which objects will the system need to hold information about, which will it need to interact with, which will remain entirely in the user domain.

The scope of the proposed system is determined by the extent of support for the work roles and by the list of objects the system will need to support. In addition, the scope of the system is reviewed from the point of view of each of the major stakeholders to identify whether their needs will be met. The list of requirements also is reviewed from the viewpoint of each stakeholder.

Once the scope of the system is decided and the list of requirements reviewed, the team is asked to identify and agree on usability targets for the proposed system. The process involves the matching of the current situation with the proposed situation and specifying those areas where usability issues may be a problem. For example, the 'gap' between possessed and required knowledge to undertake a role or a specific task within a role may highlight a potential usability problem. The usability is specified in 5 parts. The user (who), the specified activity (doing what), location (where), the targeted performance (ideal, worst case and best case), and measuring instrument (eg a benchmark) are all recorded in a usability specification table. In addition notes should be made about when in the development cycle the usability evaluation should take place (eg. early at the prototyping stage or late at the installation stage). It is important that usabilty does not only address the end-users but also considers the facilitating users (eg installers and maintenance engineers).

The outcome from this stage is an 'Initial Requirements Document' containing an agreed set of requirements for the proposed system. In addition the software designer will have gained a thorough understanding of the users.

The Initial Requirements Document is the major output from stage 2 of CRC. Each part is aimed at a specific set of stakeholders and covers a distinct set of issues. Table 3 below shows a list of contents.

1. Management Summary
   (including the business case and a brief description of the proposed system)

2. The Human Requirements
   - 2.1 Description of the objectives of the commissioning organisation
   - 2.2 List of the stakeholders together with their objectives
   - 2.3 List of key workgroups and users and their objectives

3. The High Level Functional Requirements
   - 3.1 List of work roles to be supported and why

3.2 Description of each work role in terms of users, objects and tasks

4. The Detailed Functional Requirements
   4.1 Consolidated list of objects to be supported
   4.2 Descriptions of each object together with details of user tasks associated with each object

5. The Quality Attributes
   Quality attributes may include usability, reliability, portability, performance, security, maintainability, acceptability depending on the proposed system.

6. Organisation and User Assistance Requirements
   6.1 Documentation requirements
   6.2 Training requirements
   6.3 User support
   6.4 Human computer interface requirements

7. The Technological Requirements and Constraints
   7.1 Known hardware requirements (user or supplier)
   7.2 Known software constraints (user or supplier)

table 3 List of Contents of the Initial Requirements Document

This Initial Requirements Document is produced in draft form as part of CRC stage 2. The document represents a statement of the scope of the proposed system, which will then need to be validated by reviewing its contents with the stakeholders.

Validate Scope with Stakeholders
The scope of the proposed system should be validated. A range of techniques could be used at the point depending on the type of system under consideration. Appropriate techniques might include use of questionnaires, interviewing users, building mock-ups, throw-away prototypes or holding focus groups.

Once the scope of the system has been agreed and documented, resources, timescales, tasks, milestones and deliverables can be evolved and the software designer can then proceed with the detailed design.

In addition to providing a group session approach to requirements CRC also contributes to the following from the ‘wish list’.

- supports articulation of the product concept
- supports documentation of requirements
- helps develop elevant structures on the users’ present work
- helps develop visions and design proposals

- supports identification of requirements for generic products
- is capable of working alongside market analysis techniques
- supports generic descriptions of typical users and groups of users
- supports identification and description of current workpractices
- supports identification of constraints such as cost, time and security
- supports identification and specification of acceptance criteria
- supports identification of organisational objectives, of key stakeholders and their objectives, and of key workgroups and their objectives
- supports identification of work roles to be supported and why, and descriptions of each work role, and functional requirements to support each work role
- supports identification and specification of quality attributes: usability,
- supports identification and specification of requirements for user documentation, requirements for training, requirements for user support
- supports identification and description of human computer interface requirements

Appendix B contains a ‘user guide’ to CRC Stage 1

4.9 Summary

Structural summary.....