Improving Communication and Decision Making within Quality Function Deployment [1]

Stephan Jacobs, Stefanie Kethers
Informatik V, RWTH Aachen
Ahornstr.55, 52056 Aachen, Germany
{jacobs, kethers}@informatik.rwth-aachen.de

Abstract

One of the main problems within concurrent engineering is the visibility of activities and knowledge across perspectives. Quality Function Deployment (QFD) is a method to enhance communication between developers of different views. It is based on reconciled planning and communication procedures arranged around a form called House-of-Quality (HoQ). Descriptions like "Listening to the Voice of the Customer" or "A Method for Guaranteeing Communication" emphasize the importance of communication. Most of todays QFD implementations neglect its communication aspects. Improving QFD with approaches used in the area of Computer Supported Cooperative Work (CSCW) as well as Group Decision Support Systems (GDSS) are leading to a new generation of QFD tools. This paper introduces the integration of two techniques Argumentation Systems and Electronic Meeting Systems into a QFD-tool.

1. Introduction

One of the main problems within concurrent engineering is the visibility of activities and knowledge across perspectives. Quality Function Deployment (QFD) is a method to enhance communication between developers of different views. It is based on reconciled planning and communication procedures arranged around a form called House-of-Quality (HoQ). Surprisingly most of today's QFD-sessions are graphically supported by paper and pencil. As changes, cuttings, refinements etc. lead to garbled HoQ-forms, the first generation of QFD software concentrated on editing facilities. So todays QFD implementations are mostly form editors for the HoQ. However reducing QFD in this way neglects its communication aspects. On the other hand descriptions like "Listening to the Voice of the Customer" or "A Method for Guaranteeing Communication" emphasize the importance of communication and teamwork. Moreover, the mapping of customers needs and wishes into the product design and further into the production process can be seen as a complex decision process. Improving QFD with approaches used in the area of Computer Supported Cooperative Work (CSCW) as well as Group Decision Support Systems (GDSS) is leading to a new generation of QFD tools.

This paper introduces the integration of two techniques

- Argumentation System and
- Electronic Meeting System

into a QFD-tool. We illustrate the system by first describing the shortcomings of current QFD, second outlining the difficulties to change form paper and pencil to computer supported HoQs, and third integrate systematically cooperation techniques.

Chapter 2 briefly introduces Quality Function Deployment and the House of Quality. Chapter 3 introduces the concepts and taxonomies of CSCW. Difficulties when integrating CSCW techniques into other methods are discussed. Chapter 4 analyzes QFD in more detail in terms of the QFD-product (the House of Quality), the group which performs QFD and the QFD-process.

2. Quality Function Deployment

Quality Function Deployment has been successfully employed for about 20 years. In 1983, Professor Akao presented a paper on QFD at a conference in Chicago. Since then, several joint ventures between Japanese and American companies have disseminated the method. The American Supplier Institute developed a variant of QFD that fitted into the traditional product development process. In particular, the quality tables proposed by Akao were changed into so-called *houses of quality* (cf. figure 1), mirroring the phases to be taken.

The goal of QFD is to ensure that the customers voice drives all actions concerned with quality. With customer requirements at the basis of the whole process, technical characteristics, component characteristics, process plan, and operating instructions are being deployed in four phases. To achieve this aim, the employees strive

1 This work is partially funded within the project KI-NRW by the state Nordrhein-Westfalen, Germany

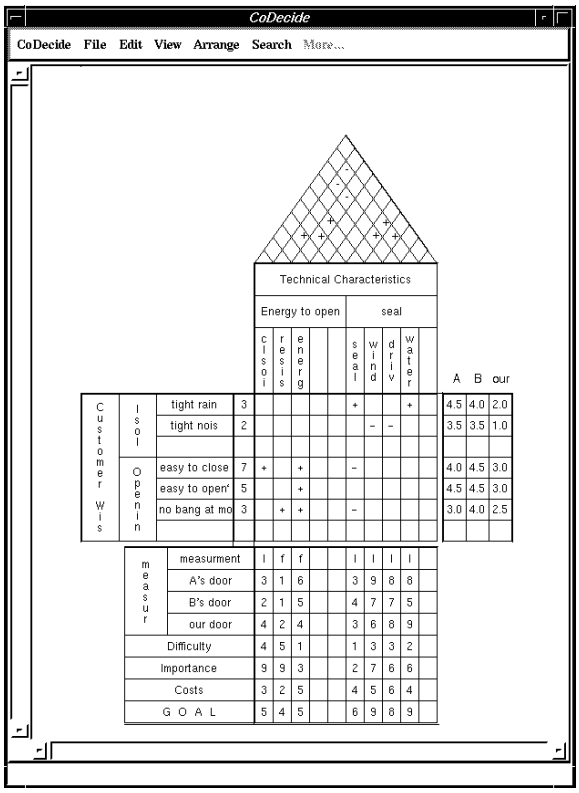

Figure 1 CoDecide – Supporting Cooperation within the House of Quality

to come to an agreement concerning their actions. This fact is an important concept in the philosophy underlying QFD. Traditional QFD is split up into four iterative phases, namely product planning, parts deployment, process & control planning, and production planning [48]. Basically all phases are similar. We have concentrated on the first and most vital of the four phases, i.e. product planning. Any errors or ommissions occurring during product planning are likely to cause great problems and costs at later stages.

3. Computer Supported Cooperative Work

QFD provides a systematic approach to cooperation between cross functional teams. However, support for cooperation is only provided by means of traditional tools, e.g. clipboard, conference room. On the other hand the growing field of Computer Supported Cooperative Work (CSCW) tends to enhance basic cooperation techniques through adding computer technology. Combining the systematic approach of cooperation within QFD with the technical oriented cooperation possibilities of CSCW should lead to considerable synergy effects. The problem

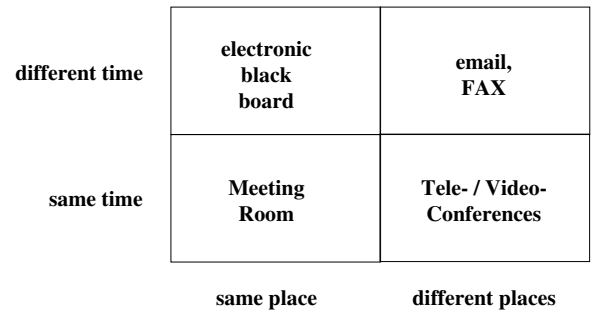

Figure 2 The Time-Space-Matrix of CSCW [27]

is to identify starting points for an integration. CSCW tends to develop general techniques. Most current CSCW-applications do not support cooperation within any traditional method so end users have to decide which of the various tools and techniques is most appropriate for their goals. Frameworks and classification schemata are developed to offer the possibility for users to pick up an appropriate CSCW technic.

3.1 Time-Space-Matrix

Frameworks to describe the field of CSCW are mostly technical oriented. The most common taxonomy is based on the notion of time and space. The *time-space-matrix* [27] (cf. figure 2) defines four categories to classify CSCW-applications. An electronic blackboard e.g. is located within the upper right cell as it is used locally but asynchronous. E-mail, the most successful groupware, is placed in the upper left cell as it enables people to communicate asynchronously from different places. However, the matrix is mostly used by groupware designers to classify their own work. It is not well suited for end users who want to adopt a specific technology to tackle their cooperation problems. Teamwork changes continuously between local and remote and between synchronous and asynchronous. The conceptualization according to the time-place-matrix destroys this continuity.

3.2 Classes of CSCW systems

Another, more application oriented survey of CSCW systems is presented by Rodden [40]. He identifies four classes of CSCW systems namely *Message Systems*, *Computer Conferencing*, *Electronic Meeting Rooms*, and *Co-Authoring Systems* (cf. figure 3).

The most familiar examples of groupware are computer based messages systems supporting the asynchronous exchange of textual messages between and within groups. The main idea behind messages systems is to support cooperation by exchange of messages. Communication

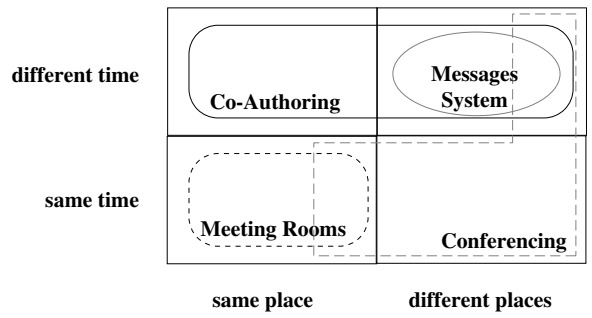

Figure 3 Application oriented classification of CSCW systems [40]

using a messages system is augmented by using special communication structures, e.g. dervied from the theory of speech-acts [53], or by adding "intelligence" to the message system to automatically route incoming messages into specific folders. Well-known messages systems include e.g. Coordinator [53], InformationLens [30], Domino [29], and Cosmos [52]. Workflow Management Systems (cf. [11], [34], [5]) are the todays most popular representatives of messages systems.

Conferencing Systems provide three different conferencing styles to the users, namely *Real Time Computer Conferencing*, *Computer Teleconferencing*, and *Desktop Conferencing*. Real Time Conferences allow a group of people who are interconnected to interact synchronously through their terminals. People communicate through an audio link and additionally can have a shared view on a common application like a group editor. Examples are RealTimeCalendar [43] and the Cognoter [14], [47]. The most popular representatives of Computer Teleconferencing are *Video Conferences*. VideoConferences sometimes require special conference rooms with appropriate group monitors and trained operators. They demand huge amount of communication so an additional video link is installed between the participants of the conference. Well-known systems are e.g. Cavecat [32] and CRUISER [42]. Desktop Conferencing combines the advantages of Video and Computer Conferencing. The video image of the conference participants are displayed on the terminal combined with the possibility to commonly work on a shared application. Examples are the MMConf System [6] and the TeamWorkStation [24].

Electronic Meeting Rooms (EMSs) and Group Decision Support Systems (GDSSs) provide computer based facilities to enhance solving of complex and unstructured problems. The goal is to improve the quality of the decision as speeding-up the decision process. Means to satisfy these goals are e.g. tools for generating ideas, techniques to structure issues or multi-criteria-decision-methods. EMSs are implemented as large conference rooms containing networked workstations which provide the tools to support decision making. Often the Meetings have to be leaded by a professional facilitator. Well-known EMSs are e.g. the Collaborative Management Room [35], the CATeam Room [12], or the CaptureLab [31].

Coauthoring systems are built for groups of authors. They support the common writing of books, specifications, programs, manuals, etc. Basics for coauthoring systems is the structuring of documents into chapters, sections, or procedures. As hypertext-systems have a strong emphasize on the structure of the text, they are well suited as a fundament for coauthoring systems. Coauthoring systems support the partitioning and merging of text. Well-known examples are Quilt [13], Co-Author [21], Grove [10], or Sepia [20].

3.3 Three types of cooperation

Another more basically application oriented taxonomy is presented by DeMichelis [8] and Schäl [44]. Three types of cooperation are identified: coordination, collaboration, and co-decision.

Coordination is a cooperative process where the users have to synchronize their individual actions because of the use of common resources. Successful coordination often depends on the proper breaking down of processes into appropriate actions [44].

The second type of cooperation is *collaboration*. Users work together on *one* task, they perform one action together. The success of a collaborative action depends on how the people share a common knowledge. Another important element is the joint understanding of a goal. When a collaborative action is finished the individual contributions to the final result remain unclear.

Decision making is one of the fundamental activities in managerial work [44]. Making a common decision is the third form of cooperation. Creating a common understanding is an important aspect of co-decision. The different types of GDSSs supporting, generation, structuring, and communication on ideas as well as voting and implementation of decisions show the wide range of different actions to be performed during the decision process.

Of course, most systems are not pure representatives of one form of cooperation. Real world systems combine of aspects coordination, collaboration, and co-decision. E.g. brainstorming tools which are used for idea generation, include ideas of collaboration and co-decision.

The classification schema by Rodden and the three types of cooperation by DeMichelis and Schäl complement one another. Rodden classifies tools and techniques whereas DeMichelis and Schäl describe basic types of cooperation independent of technology. The two schemata can easily be mapped. Message systems perform aid for coordination. As the result of collaboration is most of-

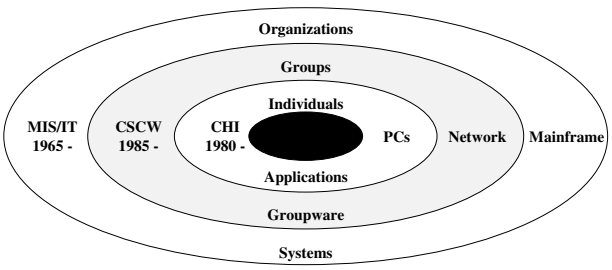

Figure 4 Development and research contexts of computer support [19]

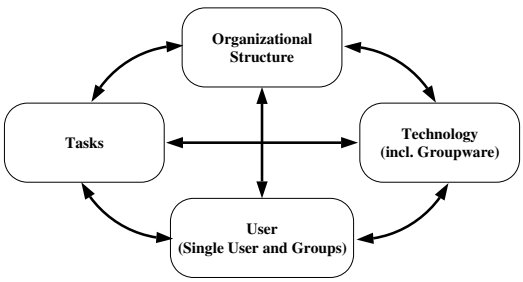

Figure 5 Specialized Leavitt-Rhomb [36]

ten some kind of text (e.g. a program, keywords on a blackboard, etc.) co-authoring systems are the best-known representatives of collaboration tools. Meeting Rooms and Conferencing Systems implement demands of co-decision, the third type of cooperation.

3.4 Finding appropriate CSCW methods

Taking a view back on QFD all basic types of cooperation can be used. The whole QFD process with filling the different parts of the HoQ form can be supported using a coordination system. Collaboration takes place while filling the HoQ. Commonly writing in the form could be enhanced by a coauthoring system. At least, QFD is performed through several meetings, where decisions are made. Meeting systems or conference tools could help to provide this aspect of QFD.

After all, the classification schemata give no proper hint, how to combine CSCW and QFD. Moreover, looking at the literature which critically explains CSCW (e.g. [19], [1], [36], [33], [18], [17]) it becomes obvious that first a very good understanding of the problem domain has to be worked out before the use of groupware has any chance of success. In comparison to traditional computer support of organizations and individuals (cf. figure 4) understanding of the domains context becomes more critical in CSCW.

Systems designed to support large organizations tend to satisfy organizational goals defined and enforced by the management. Although there is frequently no full agreement on these goals, each organizations employee is asked to use the system. So, the basis for these organization wide systems is the management decision. In the 80s the spreading of personal computers led to a huge market of software designed for individual users. Supply and demand has created successful products like word processors and spreadsheets which fulfill the individual demand of single users. Foundation for the personal computers was the satisfaction of the individuum. As users cannot be enforced by the management to successfully use groupware, success can only reached by satisfying the demands of groups. I.e., a better understanding of the group and its social contexts is necessary.

Several frameworks for analyzing the context of information systems have been proposed (e.g. KADS-II [51], [7], the Four Worlds [26]). They all do not only want to model the "pure" information system but its context in terms of "organisation", "actor", "user', "task", etc. A framework especially designed for groupware is the Specialized Leavitt-Rhomb by Oberquelle (cf. figure 5).

The rhomb indicates that there are four elements which interact when institutionalizing groupware into an organization: technology including groupware, the tasks to be performed, users, which include single users as well as groups, and the organizational structure. Even if these four elements are analyzed dispersed they are all webbed and interdependent. Changing one element will influence the others. E.g. introducing email into an enterprise can change the flow of information. Therefore the organizational hierarchy is made doubtful. As email enhances the amount of communication, user can work much closer which changes their personal interrelationships. Using email to coordinate tasks more flexible leads to a more effective way of work. For that more complex tasks can be tackled. I.e. changing one element will influence the other ones. Vice versa, if not all elements are regarded this can lead to rejection of new ideas. E.g. technology has to be well suited not only for a specific task but even for the organization and the users who have to use the technology. There might be a task of managerial decision making and the appropriate GDSS is found. But as managers are not familiar to deal with terminals, because they normally use paper, pencil, and chalkboards for representing ideas, the system will fail. Generally speaking, if groupware (systems) are designed only with the task in mind but neglecting user and organizational context, they will tend to fail.

Grudin [19] offers some additional hints where to look for typical shortcomings of groupware. Beneath technical demands like networking, (technical) communication, and (technical) coordination, and individual demands like interface design, he points out some challenges on group-

Disparity in work and benefit. Groupware applications often require additional work from individuals who do not perceive a direct benefit from the use of the application
Critical mass and Prisoners dilemma problems. Groupware may not enlist the "critical mass" of users required to be useful, or can fail because it is never to any one individual's advantage to use it.
Disruption of social processes. Groupware can lead to activity that violates social taboos threatens existing political structures, or otherwise demotivates users crucial to its success.
Exception handling. Groupware may not accomodate the wide range of exception handling and improvisation that characterizes much group activity.
• • •

Table 1 Challenges for groupware developers [19]

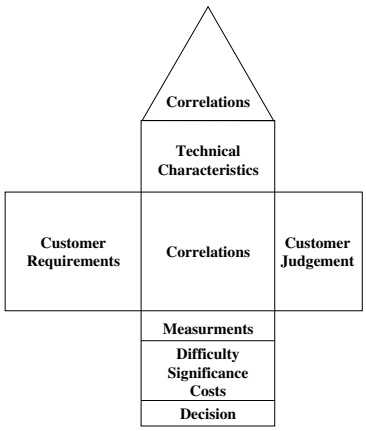

Figure 6 The eight matrices of the House of Quality

ware which are based on the social dynamics of groups (cf. table 1).

One way to successfully design groupware is to look for accepted single-user tools which can be used in group settings (e.g. wordprocessing used as a notebook in meetings). These tools have to be extended cautiously by groupware features like communication technology, shared workspaces, collaborative writing features, etc. Another way is to look for traditional methods performed by groups (e.g. brainstorming or QFD) and analyze them for shortcomings which could be tackled by some kind of groupware. Proceeding in such a way has the advantage, that users are familiar with the original method and can concentrate on technical changes. Additionally, they can compare the traditional and the supported way of working and give hints to benefits and shortcomings which can guide the designer to further improvements.

In the following chapter we will analyze QFD in terms of the process, the product, and group characteristics. We will offer three improvements which can be added independently to QFD.

4. Analysis of QFD

In this chapter QFD is analyzed taking three different views. First the QFD-product is observed and modelled. Second, the groups performing the QFD-method is examined. Third, the process is analyzed. These aspects are subsequently used to describe QFD's shortcomings, possibilities to tackle the shortcomings coupled with a discussion on advantage and disadvantages.

4.1 The QFD-Product

The product of QFD is the House-of Quality. The House of Quality consists of eight matrices which are represented in an abstract form in figure 6.

Traditionally the House of Quality is filled with paper and pencil. Assuming that an average HoQ consists of about 100 customer requirements and 150 technical characteristics [22] which leads to more than 10.000 items to fill, clarity will be lost in large projects. The great advantage of the HoQ, it's possibility to clearly show complex relationships, is limited by the number of requirements and characteristics. The attempt to break down the house in various subhouses offers only restricted improvement. Additionally, paper and pencil methods have great disadvantages when changing and updating. Insertion of new requirements, which involves insertion of 150 correlations in the inner matrix, is rather difficult, time consuming, and leads to "dirty" forms. It is impossible to perform spontaneous changes which leads to blocking of ideas and comments [50]. Another obstacle is the impossibility to quickly return to an earlier state when identifying a dead end or to just "undo" one step back.

These disadvantages urge to change the media from paper and pencil to computer, which tackles all the above mentioned shortcomings. Other advantages, which could be gained by changing to a computer are the possibilities to better divide and merge parts of the HoQ. Import and export functions could offer the possibilities to easily reuse work which was done when designing a similar product. Interfaces to CAD tools could be defined to automatically transform results from other tools into the HoQ and vice versa [25].

However, there are some losses when changing from paper to terminal. Oversized papers (clipboards) and chalkboards have been used for a long time as a group medium for discussion. As paper is "informal" people have the possibility to just outline ideas. It is well known from radiology that doctors prefer to hold röntgenograms in their hand to look and *feel* the picture. Displaying

röntgenograms on terminals is not yet accepted. Ethnographic studies [16], [46], [23], [2], show, that artifacts which seem to be irrelevant at the first sight, are significant for success or failure of products. So, it has to be examined, how different media like oversized terminals, networked terminals, displays on projectors are an appropriate discussion medium for QFD. Possible user interfaces like touch screens, window systems etc. have to be inspected. It has to be checked, which features of an QFD implementation are used and which are not. It has to be investigated how a computer supported QFD changes social dynamics of the team. E.g. how does the difference of chalk and keyboard changes the behavior of a dominant user? How does changing from a centralized meeting to networked terminals mutate communication and cooperation within QFD?

Comparing possible benefits and deficiencies (cf. table 2) the benefits of a QFD implementation suggests changing to computer supported QFD. Computer support to QFD is introduced stepwise:

- First, basic functionalities like writing, insertion of columns and rows etc. are introduced. These functionalities provide a more flexible dealing with the HoQ. The HoQ is represented using one terminal enhanced by a overhead display. The QFD session is managed (like traditional QFD meetings) by a facilitator who serves the keyboard.
- Second, interfaces to other CAD and specification tools are defined. Mapping rules transform already existing specifications into a QFD-like representation. E.g. the recursive structure of DataFlowDiagrams produces a tree. Each process can be used in the HoQ as a technical characteristic. Bringing face-to-face customer requirements and old specifications stimulates reuse. In the context of CE this is interesting as the HoQ enables the user of different phases to resolve their individual views. (This step is *not* described in this paper. Cf. [25].)
- Third, the QFD partially changes from a centralized meeting to computer conferences. Interconnected workstations provide concurrent working. Each participant can concentrate on the task he is responsible for. It has to be examined, which part of the QFD process is well suited for (asynchronous) computer conferencing and which part is better performed using the classical face-to-face meeting style (cf. section 4.3)

So far, QFDs product remains unchanged. Using a HoQ-editor instead of paper and pencil just changes the implementation or presentation of the HoQ. After all, there are some parts of the product which can be enhanced. The HoQ represents decisions. Every correlation, every customer requirement, and every technical characteristic is

+	unrestricted number of items
+	flexible update and undo
+	flexible insert, change, delete operations
+	graphical operations: contract and expand
+	blocking
+	reuse -> import / export features
+	automatic interfaces to CAD tools
?	changed social dynamics
-	changed ouptput media
-	changed input interface

Table 2 Computer supported QFD: Gains and losses

the outcome of a decision process. Often decisions are made after a controvers discussion. Studying processes with group support in mind the notion of decision is necessary as this term allows to model aspects of user involvement which cannot be expressed while just describing the outcomes of the process. This observation has led to the notion of decision-oriented processes in the domain of software and requirements engineering [9], [37], [41], [45].

The notion of explicit representation not only of the outcomes of discussion has led to the development of argumentation systems. The most popular one, Rittel's *Issue Based Information System (IBIS)* [39], distinguishes between *issues*, *positions*, and *arguments*. Relationships between these entities explore the argumentation process (cf. figure 7). Taking an issue point of view several argumentation trees are constructed with an issue as root. This tree does not reduce the discussion to its outcomes but makes the argumentation process traceable. Several current argumentation systems are based on IBIS. E.g. gIBIS [4] is a hypertext based implementation to structure policy discussions. REMAP [38] enlarges IBIS for use in requirements engineering.

The idea of IBIS can be used to represent the discussion process within QFD. IBIS is expanded be the entity deci-

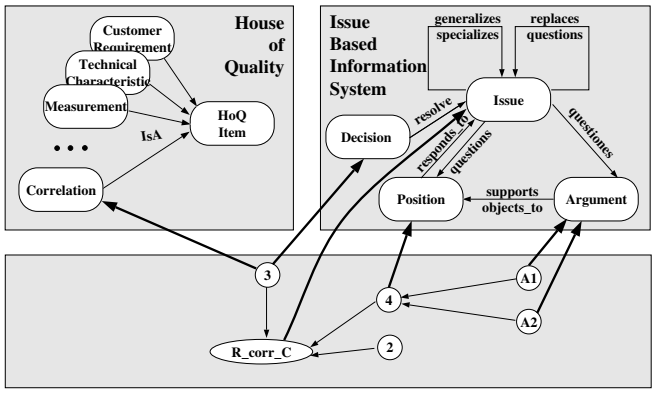

Figure 7 Formal integration of IBIS and the House of Quality

sion, which solves the issue. As decision represents the outcomes of an argumentation each item of the HoQ can be an instance of this decision. E.g. consider the correlation between requirement R and characteristic C. Some user want to have correlation 4 because of argument A1 and A2. Another user argues for correlation 2 because of argument A3 and A4. After a discussion one agrees on the average 3. Representing the argumentation is necessary to make the process traceable and reusable for other users. The product model of the HoQ is expanded by an argumentation dimension by multiple instantiation of objects (cf. figure 7). In the first steps of the QFD process the computer supported QFD offers a special discussion tool. The whole argumentation is represented using argumentation trees.

4.2 The QFD-Groups

QFD is used in several ways. Consulting firms offer QFD sessions. These firms do not need to know about the domain QFD is used for. Other firms offering e.g. information technology use QFD as a means to analyze and figure out requirements of potential customers. Some enterprises use QFD in-house. These differences make clear that the groups performing QFD differ. Some groups use QFD for the first time, others have made several experiences with QFD. Some groups are formed only for a particular QFD session, others exist for a long time. Some groups are homogeneous in terms of knowledge, position, or power, others are heterogenous. Some groups are trained to interact with computers and other technical media, other use traditional paper and pencil tools. Within customer-oriented projects the customer can personally participate in the QFD-sessions. In market-driven projects the marketing group are advocates for the customers. Moreover, group changes their behavior and dynamics over time. This is especially true for groups formed only for one QFD session. The structure of the group, i.e. the roles, is not found nor the dynamics, i.e. the way of communication and cooperation.

According to Tuckman and Jensen [49] and Cole and Nast-Cole [3] there are five stages of group development, namely forming, storming, norming, performing, and adjourning. A group constituted only for one QFD (which can last e.g. four weeks) runs through all five states. The influence of computer depends on the actual stage. E.g. in the forming stage which is determined by identification (who else is in the group? what is the task? how are we going to precede?) the computer can block the group members when becoming acquainted. Another team, working together for a long time with clearly identified roles and norms to solve conflicts and differences is in the performing stage and will rest in this stage after

Group	
attributes	open vs. closed
	formal vs. informal
	interacting vs. coacting
	cohesive vs. non-cohesive
	hierarchical vs. peers
stages	forming
	storming
	norming
	perfoming
	adjourning
Individuals	
computer	laptop, networked PC, stand-alone
experiences:	CAD-workstation, no experience, etc.
department:	management, production, design, marketing, ...

Table 3 Framework for characteristics of groups and individuals

the QFD session. The norms and dynamics chosen by the group could be supported by computer technology.

The way a particular group will perform QFD hardly depends on the characteristics, experiences, and abilities of the group. For that each time a new QFD session is performed these group aspects have to be thought about again. Of course, it is not possible to formally describe the group, develop specific strategies for all kind of groups, and by that guarantee a successful QFD. However, it is possible to offer frameworks containing several attributes (cf. table 3). These attributes can be used as a checklist to classify the group. The medium used to represent the HoQ should be appropriate to the specific group. E.g. a group with only little experience with computers will not accept networked workstations as a representation medium for the HoQ. In this case, the HoQ should be developed with pen and pencil. However, a computer supported HoQ-editor could be used by a recorder. Regularly plotted posters of the actual version of the HoQ could be used to enhance management of versions or reduce the inconvenience of removals and updates. A group of experienced computer users, familiar with using computers as a communication medium (e-mail, usenet, etc.), can use a computer supported HoQ on networked workstation. Asynchronous discussions on the workstations can precede the main QFD session. Managers who are familiar to use laptops during meeting, can interact with the HoQ through their laptop. Individual querying about details and rationales behind stated requirements can be supported.

4.3 The QFD-Process

In the following we concentrate on the first phase of QFD, the product planning. As the other phases are

Step	Task	Joining Groups
Determine customer requirements	Analyze marketing informations	Marketing
Determine technical characteristics	Analyze customer requirements and define technical characteristics	Marketing, Design
Fix correlation matrix	Find relations between characteristics and requirements	Marketing, Design
Fix roof matrix	Find technical relations, estimate concepts	Design
Comparison of competitive products by the customers	Analyze information about the market	Marketing, Design
Comparison of competitive products	Analyze technical information	Design
Determine importance and difficulty	Determine importance and difficulty of technical characteristics	Marketing, Design
Detemine goal values	Decide which characteristics have to be reached	Marketing, Design Management

Table 4 QFD steps

similar, the observations and statements concerning this phase can easily be adapted to the other phases.

The product planning process is performed around the House of Quality. The QFD team has to fill each of the matrices of the HoQ with the appropriate data. Figure 1 shows an example house of quality after these eight steps have been completed. Most of these steps rely on profound knowledge from various fields, such as marketing, design, production, etc. Hence, they can only be performed by interdisciplinary teams, whose members each possess some part of the necessary knowledge. Each of the steps is briefly described in table 4. Additionally the participating groups are mentioned for an example QFD. Of course, these steps are not performed iteratively. Like other planning and designing processes it can be characterized by iterations, backtracking, etc.

Experiences with real QFD projects show, that participants spend up to 40–60 hours within sessions [15]. This observation emphasizes the crucial part of sessions within QFD. On the other hand field studies demonstrate that sessions are often ineffective. In [35] several of these shortcomings, like blocking or free riding, inherent to traditional sessions are examined. GDSSs tackle these deficiencies and make sessions more effective in terms of time and satisfaction.

The eight steps listed in table 4 describe the QFD process only on a very coarse level. In detail each of the steps has finer granularity. According to [28] each session is embedded in a *starting*, a *coordination*, an *asynchronous preparing*, a *real synchronous session*, and a *stop phase*. Traditional QFD focuses on the synchronous session. For each entry in one of the matrices at last a brief discussion takes place in the session. As not each participant is interested in every detail the sessions consume unnecessary time. Preceding the session with an asynchronous phase can anticipate several "small" discussions. Thus, the session team is able to concentrate on the essential decisions.

The step *fix roof matrix* is chosen to be preceded by a computer supported asynchronous discussion because only technical experienced designer participate in this session. The HoQ is used as a blackboard, i.e. each designer can access to the HoQ from his workstation. There is no schedule but a deadline when he can enter his proposals and requirements into the HoQ. He can enter comments to other items, and by this start a discussion. Conflicts based on misunderstandings like synonyms, homonyms, misinterpreting or unclear statements can be clarified during this phase. A meeting schedule based on actual HoQ can be developed. The actual HoQ can be used as starting point for the following synchronous discussion.

5. CoDecide – Supporting cooperation within the House-of-Quality

Actually, we have implemented a House-of-Quality tool [28]. Although our interest is mainly focused on cooperation support, the development of the HoQ-form-editor is important for the success of the tool, as it is the interface between human and computer. The editor is based on the X11 window system. Thus, there ia a look-and-feel integration of the HoQ into a working environment (cf. figure 1). Cut and paste functionalities between the HoQ and other applications are supported. CoDecide is based on a database. So, sharing of information and recording of HoQ is supported by database technology. The actual input medium if the keyboard, which can be used in sessions by a recorder. The output medium is the monitor. To enhance sessions the content of the screen can be displayed with a overhead projector. CoDecide enables to import and export parts of the HoQ. This function can be used to support reuse, or to put different specifications face-to-face.

The basic database allows different users to share and merge their information. So, sessions can be prepared distributed and asynchronous. Discussions are enhanced with a IBIS-oriented argumentation editor. Conflicting positions, arguments supporting and oponing a position are recorded to make the decisions (i.e. the entries in the HoQ) traceable. Another possibility to assist mutual understanding is to fit comments and annotations to entries. Thus, users are able to explain their ideas and contributions more clearly. Coordination of the whole QFD-process is supported by a task-management tool. The tool enables, to define responsibilites, deadlines, or schedules and synchronizes the differnt tasks.

6. Summary and outlook

We have shown that QFD is well suited to be supported by cooperation technologies offered through CSCW. We have outlined a concept how techniques and methods to support cooperation can be used to enhance QFD. Parts of the concept have been implemented [28]. Currently we are evaluating the tool as well as the concepts within the domain of factory layout. The House-of-Quality is used twofold. First, as a medium to express the wishes of the customers. The HoQ supports preparing the sessions, clarifying the requirements concerning the factory and helps to structure the discussion. Second, the HoQ is used in-house. Interfaces to CAD-tools to outline the factories enable the user to concurrently develop the layout and to express changes and updates automatically in the HoQ. The clear form can be used to estimate the changes according to the stated requirements. By that, combining HoQ and CAD-tools shortens the distance between decision making and implementation.

7. References

[1] L.J. Bannon and K. Schmidt. Four Characters in the Search of a Context. In J.M. Bowers and S.D. Benford, editors, *Studies in Computer Supported Cooperative Work*, pages 3–16. North-Holland, 1991.

[2] Richard Bentley, Tom Rodden, Peter Sawyer, Ian Sommerville, John Hughes, David Randall, and Dan Shapiro. Ethnographically-Informed Systems Design for Air Traffic Control. In *Proceedings of the Conference on Computer Supported Cooperative Work, CSCW'92, Toronto*, pages 123–129, 11 1992.

[3] Paul Cole and Judith Nast-Cole. A Primer on Group Dynamics for Groupware Developers. In David Marca and Geoffrey Bock, editors, *Groupware: Software for Compute-Supported Cooperative Work*, pages 44–59. IEEE Computer Society Press, 1992.

[4] Jeff Conklin and Michael L. Begemann. gIBIS: A Hypertext Tool for Exploratoy Policy Discussion. *ACM TOOIS*, pages 140–151, 10 1988.

[5] SODAN Consultancy. Workflow Management Products. Technical report, SODAN, 11 1992.

[6] Terrence Crowley, Paul Milazzo, Ellie Baker, Harry Forsdick, and Raymond Tomlinson. MMConf: An Infrastructure for Building Shared Multimedia Applications. In *Proceedings of the Conference on Computer Supported Cooperative Work CSCW'90*, pages 329–342, 10 1990.

[7] Robert de Hoog, Rob Martil, Bob Wielinga, Rob Taylor, and Clive Bright. The Common KADS model set. Technical report, ESPRIT P 5248, 1992.

[8] G. DeMichelis. *Computer Support for Cooperative Work*. Butler Cox Foundation, 1990.

[9] Mark Dowson. Integrated Project Support with IStar. *IEEE Software*, pages 6–15, 11 1987.

[10] Clarence Ellis, Simon Gibbs, and Gail Rein. Groupware. *Communications of the ACM*, 34(1):39–58, 1 1991.

[11] Clarence Ellis and Gary Nutt. Modelling and Enactment of Workflow Systems. In *Proceedings of the Conference on Application and Theory of Petri-Nets*, pages 1–16. Springer, Lecture Notes 691, 1993.

[12] T. Ferwagner, Y. Wang, H. Lewe, and H. Krcmar. Experiences in Designing the Hohenheim CATeam Room. In S.D. Benford J.M. Bowers, editor, *Studies in Computer Supported Cooperative Work*, pages 251–266. North-Holland, 1991.

[13] R.S. Fish, R.E. Kraut, M.D. Leland, and M. Cohen. Quilt: A Collaborative Tool for Cooperative Writing. In *Proceedings of the Conference on Office Information Systems COIS'88, Palo Alto*, 1988.

[14] Gregg Foster and Mark Stefik. Cognoter. In *Proceedings of the Conference on Computer Supported Cooperative Work, CSCW, Austin*, pages 7–15, 1986.

[15] Gfmt. Managementtraining Qualität. Technical report, GFMT München, 1992.

[16] Joseph Goguen and Charlotte Linde. Techniques for Requirements Elicitation. In *Proceedings of RE93, IEEE International Symposium on Requirements Engineering, San Diego, California*, pages 152–164, 1 1993.

[17] Jonathan Grudin. Why CSCW Applications Fail: Problems in the Design and Evaluation of Organizational Systems. In *Proceedings of the Conference on Computer Supported Cooperative Work, CSCW'88, Portland*, pages 85–93, 1988.

[18] Jonathan Grudin. Why Groupware Applications Fail: Problems in the Design and Evaluation. *Office: Technology and People*, 4(3):245–264, 1989.

[19] Jonathan Grudin. Groupware and Social Dynamics: Eight Challenges for Developers. *Communications of the ACM*, 37(1):93–105, 1 1994.

[20] J. Haake and B. Wilson. Supporting Collaborative Writing of Hyperdocuments in SEPIA. In *Proceedings of the Conference on Computer Supported Cooperative Work, Toronto*, pages 138–147, 1992.

[21] U. Hahn, M. Jarke, S. Eherer, and K. Kreplin. CoAuthor - A Hypermedia Group Authoring System. In J.M. Bowers and S.D. Benford, editors, *Studies in Computer Supported Cooperative Work*, pages 79–100. North-Holland, 1991.

[22] J.R. Hauser and D. Clausing. The House of Quality. *Harvard Business Review*, pages 63–73, 5 1988.

[23] Christian Heath and Paul Luff. Collaboration and Control. Crisis Management and Multimedia Technology in London Underground Line Control Rooms. *Computer Supported Cooperative Work*, 1(1-2):69–94, 1992.

[24] Hiroshi Ishii. TeamWorkStation: Towards a Seamless Shared Workspace. In *Proceedings of the CSCW'90, Los Angeles*, pages 13–26, 10 1990.

[25] Stephan Jacobs and Klaus Pohl. Traceability between Cross-Functional Teams. In *First International Conference*

on Concurrent Engineering, Research and Applications, CERA'94, Pittsburgh, Pennsylvania, 8 1994.

[26] Matthias Jarke and Klaus Pohl. Establishing Vision in Context: Towards a Model of Requirements Processes. *Proc. of the 14th Int. Conf. on Information Systems, Dec. 5-8, Orlando, Florida, USA*, 1993.

[27] R. Johansen. *Groupware: Computer Support for Business Teams*. The Free Press, New York, 1988.

[28] Stefanie Kethers. Aufbau einer Arbeitsumgebung zur kooperativen Erstellung von Qualitätsplänen im Rahmen der Produktplanung, Diplomacy Thesis, RWTH Aachen, 1993.

[29] T. Kreifelts and H. Wötzel. Distribution and Error Handling in an Office Procedure System. In *Proceedings IFIP WG 8.4 Working Conference on Office Systems Methods and Tools, Pisa*, pages 197–208, 1986.

[30] Thomas W. Malone, Kenneth R. Grant, and Franklyn A. Turbak. The Information Lens: An Intelligent System for Information Sharing in Organizations. In *CHI/GI Proceedings '86*, pages 1–8. ACM Press, 1986.

[31] Marilyn Mantei. Capturing the Capture Lab Concepts: A Case Study in the Design of Computer Supported Meeting Environments. In *Proceedings of the Conference on Computer Supported Cooperative Work CSCW, Portland*, pages 26–29, 9 1988.

[32] Marilyn M. Mantei, Ronald M. Baecker, Abigail J. Sellen, William A.S. Buxton, and Thomas Milligan. Experience in the Use of a Media Space. In *ACM Conference on Human Factors in Computing*, pages 203–208, 3 1991.

[33] M.L. Markus and T. Connolly. Why CSCW Applications Fail: Problems in the Adoption of Interdependent Work Tools. In *Proceedings of the Conference on Computer Supported Cooperative Work, CSCW'90, Los Angeles*, pages 371–380, 1990.

[34] Raul Medina-Mora, Terry Winograd, Rodrigo Flores, and Fernando Flores. The Action Workflow Approach To Workflow Management Technology. In *Proceedings of the Conference on CSCW '92, Toronto*, pages 281–288, 11 1992.

[35] Jay Nunamaker, A.R. Dennis, J.S. Valacich, D.R. Vogel, and J.F. George. Electronic Meeting Systems to Support Group Work. *Communications of the ACM*, 34(7):40–61, 7 1991.

[36] Horst Oberquelle. CSCW- und Groupware Kritik (in German). In Horst Oberquelle, editor, *Kooperative Arbeit und Computerunterst tzung*, pages 37–62. Verlag f r Angewandte Psych., 1991.

[37] Colin Potts. A Generic Model for Representing Design Methods. *ACM Panel 3/Software Engineering Research Agendas*, pages 217–226, 1989.

[38] Balasubramaniam Ramesh and Vasant Dhar. Process Knowledge-Based Group Support for Requirements Engineering. *Transaction on Software Engineering*, 18(6), 1992.

[39] Horst W.J. Rittel and Melvin M. Webber. Dilemmas in a General Theory of Planning. *Policy Sciences*, 4:155–169, 1973.

[40] Tom Rodden. A Survey of CSCW Systems. *Interacting with Computers*, 3(3):319–353, 1991.

[41] Collette Rolland. Modeling the Requirements Engineering Process. In *Proceedings of the 3rd European-Japanese Seminar on Information Modelling and Knowledge Bases, Budapest*, 1993.

[42] R.W. Root. Design for a Multi-Media Vehicle for Social Browsing. In *Proceedings of the Conference on Computer Supported Cooperative Work CSCW, Portland*, pages 25–38, 9 1988.

[43] S. Sarin and I. Greif. Computer Based Real-Time Conferencing Systems. *IEEE Computer*, 18(10):33–45, 10 1985.

[44] Thomas Schäl. Workflow Management Technology in Complex Organizations, Doctoral Thesis. Technical report, RWTH Aachen, to be published 1994.

[45] J.R. Schmitt. Product Modelling for Requirements Engineering Process Modelling. In *Proceedings IFIP WG A-30 Information System Development Process, Como*, pages 231–246, 9 1993.

[46] Ian Sommerville, Tom Rodden, Pete Sawyer, Richard Bentley, and Michael Twidale. Integrating ethnography into the requirements engineering process. In *Proceedings of RE'93, IEEE International Symposium on Requirements Engineering, San Diego, California*, pages 165–173, 1 1993.

[47] Mark Stefik, Gregg Foster, Daniel G. Bobrow, Kenneth Kahn, Stan Lanning, and Lucy Suchman. Beyond the Chalkboard: Computer Support for Collaboration and Problem Solving in Meetings. *Transactions of the ACM*, 30(1):32–47, 1 1987.

[48] Ray Thackeray and George van Treeck. Applying Quality Function Deployment for Software Product Development. *Journal of Engineering Design*, 1(4):389–410, 1990.

[49] B. Tuckman and M.A. Jensen. Stages of Small Group Development Revisited. *Group and Organizational Studies*, 2:419–427, 1977.

[50] Joseph Valacich, Alan Dennis, and J. Nunamaker Jr. Electronic meeting support: the GroupSystems concept. In Saul Greenberg, editor, *Computer-supported Cooperative Work and Groupwork*, pages 133–154. Academic Press, 1991.

[51] B. Wielinga, A. Schreiber, and J. Breuker. KADS: A Modelling Approacch to Knowledge Engineering. In B.E. Buchanan and D.C. Wilkins, editors, *Readings in Knowledge Acquisition and Learning*, pages 92–116. Morgan Kaufmann Publishers, 1993.

[52] Sylvia Wilbur and Robert Young. *The COSMSO Project: A Multi-disciplinary Approach to Design for Computer-Supported Group Working*, pages 147–155. Elsevier (North-Holland), 1988.

[53] T. Winograd and F. Flores. *Understanding Computers and Cognition: A New Foundation for Design*. Ablex Norwood, NJ, 1986.