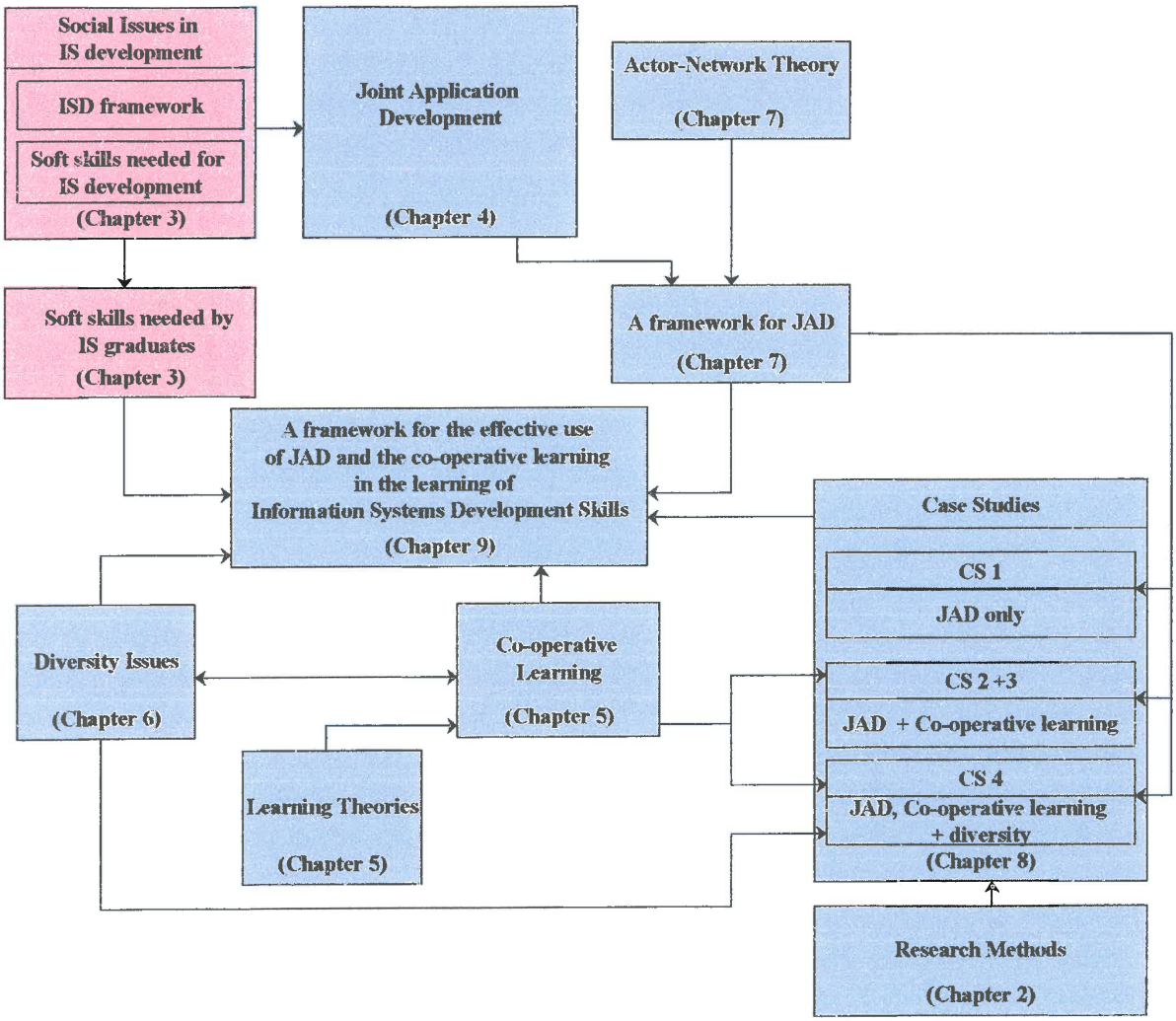
Chapter 3

Social Issues in Information Systems Development



Chapter 3

Social Issues in Information Systems Development

Information Systems can be defined as the set of technical and human resources devoted to the management of information in organisations [Ciborra, 1998]. It is thus a combination of the human with the technical. This chapter will show that the development of IS is a social process as well as a technical one.

ISD is a complex, social process which involves interaction between many stakeholders [Kirsch & Cummings, 1996]. IS professionals need not only technology skills, but also skills in business operation, management and interpersonal skills in order to cope with the world of Information Technology [Lee, Trauth & Farwell, 1995]. Dahlbom & Mathiassen [1993] suggest that systems developers need to combine a mechanistic understanding of computers with a romantic appreciation of the complexities of social issues.

This chapter gives an overview of a framework that has been proposed for ISD and shows where this research fits into this framework. This leads to a discussion of the social skills needed by IS developers and the implications of this for tertiary education.

3.1 A FRAMEWORK FOR ISD

ISD includes the analysis, design, construction and implementation of information systems. Hirschheim, Klein & Lyytinen [1996] have proposed a federated framework for ISD that emphasises the diverse nature of ISD. They claim that the framework helps us to understand the dynamics of ISD and provides us with categories for interpreting and

relating the research. The authors claim that the field of ISD is too wide to be catered for by one paradigm and that the framework is necessary in order to incorporate all the domains and orientations necessary. Hirschheim et al's [1996] paper has been termed by various authors as "seminal" and "bold" [Introna, 1996], " well-structured and significant" [Kerola, 1996], "worthwhile" and "of considerable interest" [Walsham, 1996] and "rich and suggestive" [Ang, 1996]. It is obviously important and is also fairly current and has therefore been chosen as the framework within which to place this research.

3.1.1 Behavioural orientations

Hirschheim et al. [1996] use Habermas's social action theory to divide ISD into four behavioural orientations that underlie the behaviour of the various actors during ISD. These orientations are used to capture the underlying values, goals and epistemological underpinnings that drive the development activity. These four behavioural orientations are:

- **Instrumental (control) orientation**, which is concerned with achieving the predefined end-results and treats everything in the domain as controllable objects;
- **Strategic (control) orientation,** which is concerned with achieving the predefined end-results but treats each human active in the domain as an independent conscious agent with a will of their own;
- **Communicative (sense-making) orientation,** which is concerned with achieving a common understanding through communication; and
- **Discursive (argumentation) orientation,** which is concerned with achieving clarification and justification and providing reasons and evidence.

The first two orientations emphasise control but differ in how they see people as subjects of that control. The first sees people as objects whereas the second sees people as intelligent agents. The communicative orientation emphasises creating shared meanings through sense-making. The primary emphasis in the discursive orientation is argumentation which makes sure that claims made during communication are clarified

and called into question.

A diagram, showing how the four behavioural orientations are combined with domains of change in order to create the federated framework, is given in Table 3.1 later in this section.

3.1.2 Domains of change

Etzioni's malleability hypothesis is used by Hirschheim et al. [1996] to define three domains of change in ISD. This helps to define what is being changed during the development. They identify the following three domains of change:

- **Technology**, which includes the physical means and technical know-how that are used to accomplish information processing tasks;
- **Organisation**, which includes the organised behaviour affected by the ISD, for example the work arrangements and procedures, roles, power and culture of the organisation; and
- **Language,** which is used by IS developers in the handling of symbols and is required to carry out the human transactions and co-ordinate them. Language is the medium that allows communication to take place and includes any form of symbolic representation that conveys meaning.

Kerola [1996] points out that it must be understood that all these categories change in all systems development and that it is only in their relative differences that we can identify what the focus and source of change are. This is supported by Dittrich and Floyd [1996] who say that real ISD and research must combine different views and cannot restrict themselves to one perception of what the domain of change is.

Kerola [1996] suggests that Hirschheim et al. need to add a fourth domain of change in order to cater for the development and use of information systems in societies. This fourth dimension is:

- **Actor/frame dimension**, which focuses on the change in the values and frames

of the human actors involved in the ISD.

While this sounds like a valid argument, Kerola does not analyse or explain it much further. No further articles have been found that do expand on it and it has, therefore, been left out of the federated framework presented in the following section.

3.1.3 Federated framework

The framework proposed by Hirschheim et al. [1996] cross-relates the domains and orientations to form the object system classes shown in Table 3.1.

DOMAINS	ORIENTATIONS			
	CONTROL		SENSE-MAKING	ARGUMENTA-TION
	INSTRUMENTAL	STRATEGIC	COMMUNICATIVE	DISCURSIVE
TECHNO-LOGY	Information Technology Systems			
LANGUAGE	Formalised Symbol Manipulation Systems	Manipulative Communica-tion Systems	Symbolic Interaction Systems	Systems for Rational Argumentation
ORGANISA-TION	Mechanistic Social Systems	Political Systems	Cultural Social Systems	Systems for Institutional Checks & Balances

Table 3.1: A federated framework for ISD [Hirschheim et al., 1996, p.17]

As the technology domain only consists of physical and not human artifacts, the strategic, communicative and discursive orientations are not used for this domain as they require human traits. Language and organisation, on the other hand, will always include human actors and all four orientations are appropriate for them. We are, therefore, left with nine object system classes in the original Hirschheim et al. [1996] framework.

As mentioned before, Dittrich and Floyd [1996] propose that there is some problem in trying to classify and restrict oneself to a particular domain of change and thus to a particular class. As one may have multiple domains of change and different perspectives on those domains, the development strategy will go across the boundaries of the object classes. They proposed that the object system classes are more useful for clarifying the different perspectives that one might have of a system rather than trying to match the development strategy being used to a particular class.

This is the view held by this author and is the reason why this research is placed with the sense-making (communicative) and argumentation (discursive) orientations across the language and organisation domains of change. It is also why the author has chosen to discuss from an orientation point of view in the following subsection.

Hirschheim et al. [1996] predict that although most IS efforts at the moment are directed at the top-left of their framework, this is changing and that more effort must be directed at the bottom-right part of the framework. Walsham [1996] agrees with this analysis although he objects to their use of "biassed language" that he says they use in order to support this.

3.1.4 Development strategies across the orientations

Development strategies will differ for IS, depending on the orientation and domains of change. This thesis will investigate the development strategies of the control, sense-making and argumentation orientations. The discussion will be augmented and compared with hard systems, soft systems and dialectic approaches to systems development proposed by other authors [Checkland, 1993; Checkland & Scholes, 1990; Dahlbom & Mathiassen, 1993].

3.1.4.1 Control orientation (hard systems)

Control implies that one can predict the behaviour of the system and can take corrective

action (if necessary) to make the system achieve its given purpose. The development strategies for this orientation thus assume that by setting up the criteria properly and determining all the variables that affect the system, one should be able to achieve the expected outcomes.

The key methods of development are based on engineering principles and using those principles in determining requirements, design and implementation. Determining requirements' definitions from an engineering perspective assumes that one can define the problem and determine a solution which meets the technical standards of reliability, adequate performance and cost-efficiency. One of the main aims is to find a fit between the information needs of the organisation, the task and the information system that should meet those needs. Structured design, analysis of code, data flow architectures and modelling user behaviour are important in this method [Hirschheim et al., 1996].

Development strategies must take into account who controls communication and the meaning of language. This is seen as a means of manipulating the design rather than a search for real truth. The solution looked for in the requirements definition, is often that determined by a particular group, rather than the organisation as a whole. Design is concerned with accumulating power, authority and other means of influence for personal or group advantage according to Hirschheim et al. [1996]. The communication will be distorted by this. Walsham [1996] objects to Hirschheim et al.'s bias which he says is evidenced by their use of the word "manipulative" and by their indicating that the communication is distorted in this class.

In the control orientation the IT personnel act as "experts" who know things that the users do not [Wilson, 1997]. The end-user is not seen as a conscious, reflective actor in the design process.

The control orientation has similar characteristics to hard systems thinking. Those who practise hard systems thinking expect that they will be able to get a clear and exact representation of the world. The world is seen as stable and ordered. This enables a

developer of a system to be able to get an exact picture of the problem and to be able to find an optimal solution to that problem through engineering principles [Dahlbom & Mathiassen, 1993]. The hard systems methodology assumes that an objective can be defined and the system is engineered to reach the objective [Checkland, 1993].

While the hard systems methodology has been successful in some spheres where rational human decision-making is possible, it may be more difficult to apply in the field of IS. Hard systems, as mentioned before, assumes that the problems can always be expressed and the objectives can always be defined and agreed upon. This is not always true for social systems where the problems are often "fuzzy" and difficult to define [Checkland, 1993; Schecter, 1991].

The control orientation is very important in ISD but needs to be augmented, enhanced or even replaced by strategies for sense-making and argumentation. These will be described below.

3.1.4.2 Sense-making orientation (soft systems)

Sense-making emphasises the potential role of the IS for achieving mutual understanding. It is important to find "*acceptable, understandable interpretations of ambiguous or unintelligible events which typically do not make sense within established viewpoints and policy frames of reference*" [Hirschheim et al., 1996, p.35]. All the players are confronted with the challenge of trying to understand the "*alien meaning, irrational behaviour and shifting boundaries of meaning, situations and actions*" of the other players [Hirschheim, 1996, p.46]. One of the most difficult problems in ISD is to create a shared vision with the user of what the final product should look like [Wiegers, 1993]. This can cause products to fail as they do not match the expectations of the user community.

Development methodologies for this orientation depend on our ability to converse with one another and to facilitate consensus building by contact with one another. The

emphasis is on building shared meanings and shared use of language between users and developers. Direct participation of both groups is needed. Sense-making involves the elicitation and creation of knowledge in order to make a situation clear. This is especially needed in complex, ill-structured domains [Nosek & McNeese, 1997].

From an organisational perspective, the structures must be created that facilitate formal and informal interaction, thereby allowing the sharing of opinions and observations. The emphasis here is on the roles, institutions, practices and cognitive frames of the people in the organisation.

A special focus is placed on the hidden, taken-for-granted organisational practices, as it is these that help to make sense of the organisation. Studies must be made of the organisations co-operative work practices in order to determine these. The outcomes of this strategy are to delineate these hidden and taken-for-granted practices in order to modify and mould the contexts within the organisation so that sense can be made of these practices [Hirschheim et al., 1996].

This orientation has many similarities with soft systems thinking. As human beings we cannot help but attribute different meanings and different interpretations to our world. These perceptions can lead to actions which could be effective or disastrous [Checkland & Scholes, 1990]. Soft Systems Thinking tries to get us to consider different perspectives and to learn about the world by expressing and debating those perspectives. The idea is that developers should compare the beliefs and attitudes of the different stakeholders and learn from those differences. The world is not seen as a structured, stable environment but is seen as constantly changing [Dahlbom & Mathiassen, 1993]

The Soft Systems Methodology (SSM) and its derivatives, like the Multiview methodology, are appropriate methods to be used for sense-making [Checkland, 1993; Checkland & Scholes, 1990]. They claim to be able to deal with unstructured problems and enable the developers to get different perspectives on the problem and possible

changes that might be needed to achieve success.

In their original forms these methods can be criticised for focussing on the requirements definition and design aspects of ISD, thus forcing a waterfall-type approach to systems development. Avison, Wood-Harper, Vidgen & Wood [1998] have recently proposed an amendment to Multiview, that they call Multiview2, which overcomes this problem and combines the aspects of organisational analysis, information modelling, sociotechnical analysis with software development into, what they call, an interpretive scheme. They thus cross the different domains of change defined by Hirschheim et al [1996] within the orientation of sense-making. Mediation is used to pull the different perspectives together.

Soft Systems thinking has been criticized, however, for not dealing with issues of power and social change. According to Schecter [1991] soft systems thinkers have an idealistic view of the world and assume that free, open and democratic debate among stakeholders is possible whereas, in reality, communication is often distorted by the domination of some people over others. (This is a critical systems thinking view of soft systems thinking. Critical thinking will be briefly described in the following subsection.)

3.1.4.3 Argumentation orientation (dialectic/ critical systems)

The argumentation orientation aims to achieve clarification and justification of claims and provide reasons and evidence for them. It is important to provide supporting rational argument in the design and use of IS [Hirschheim et al., 1996].

The idea of rational argument is that everyone should have the chance to put forward their argument and that they should be able to have a rational debate about the claims. Unfortunately, as Walsham [1996] suggests, the problem occurs when everyone has put forward their rational arguments and there remains disagreement as to which argument is better. The ideal situation may be difficult to achieve in real life.

When ISD is viewed as an argumentative process, then one would expect that there would not be a well-defined set of activities that can be planned and systematically carried out. As statements and solutions are made and different viewpoints are made, they are scrutinized and debated. This is an iterative process. Each solution, generated by the argument, is debated and modified and the new solutions put forward. These are again debated until consensus is reached [Koh & Heng, 1996].

The outcomes of the argumentative orientation are that the level and understanding of an argument is improved by improving the available evidence and clarifying communication breakdowns. Tools like prototyping, JAD and statistical analysis can be used to raise doubts and provide evidence. The principles involved are that all claims must be tested and that warrants for or against claims must be found [Hirschheim et al., 1996]. Critical examination and self reflection are essential elements of this orientation [Wilson, 1997].

From an organisational perspective the methodologies used should try to eliminate any distortion brought about by the hierarchical nature and other forms of power of the organisation. There should be methods to cross-check and make sure that bias and self-deception are reduced. Critical thinking is imperative.

Walsham [1996] maintains that it is infeasible to expect that one can eliminate any distorting effects of power and that power and rationality are inseparable and should be studied together. He says too that one should be careful of labelling everyone who is in a position of power as being ready to abuse that power as most senior managers do try to do the right thing. The importance of the concept of "power" is also described by Introna [1996] who sees it as something that needs to be understood and analysed when considering developing IS within the framework given. *"Power is not an obstruction forced upon ignorance or vested interests, but a relationship of inequality between human beings."* [Beirne, Ramsay & Pantelli, 1998, p.303]. It is thus inevitable that power will affect any participative design methods.

Methods of making sure that self-managed teams are democratic, unbiased and critical of their own work must be fostered. The widest possible number of stakeholders should be involved in the development process.

Dahlbom & Mathiassen [1993] suggest a more critical approach to systems development, which they call a dialectic approach. The argumentation orientation has similarities with this approach. Reality is seen as a set of contradictions which are related and dynamically changing. The world is seen as a place of chaos and conflict which we cannot really understand. When developing systems, the development team must understand and challenge existing established traditions and intervene to change them if necessary. They will need to find out what the actual practice is in the organisation rather than the method that has been defined.

While critical systems thinking cannot really be equated to the argumentative orientation or the dialectic approach, there are many points of similarity. Critical systems thinking rests on three commitments, namely [Schecter, 1991]:

- Commitment to critique, which means that practitioners are committed to questioning the methods, practice and theory of their disciplines;
- commitment to emancipation, which is commitment to a free and equal participation by all; and
- commitment to pluralism, which indicates that critical systems thinkers do not imply that hard and soft systems thinking should be done away with, but rather feel that all these approaches have a contribution to make and that one would be inadequate to deal with the complexities of systems design.

3.1.4.4 In summary

Software development should not be seen as an engineering discipline where one is merely making a product that must be reliable, efficient and cost-effective. While these factors are important, one must also realise that an Information System can change the way people work and there is thus, a complex interplay between technology and

society.

Users must be involved and developers need to develop the skills necessary to work with those users in order to achieve the best results. Dahlbom & Mathiassen [1993] maintain that the developers need to be "sociotechnical experts" who are critical about their work.

Dahlbom & Mathiassen [1993] suggest that if the problem is well structured and certain then a hard systems approach could be appropriate, but if the problem is unstructured and uncertain, then a soft systems or dialectic approach should be used. Schecter [1991] suggests a pluralistic path that recognizes each of the approaches and deals with different dimensions of the problem.

Most systems development methodologies acknowledge the need for getting the support of senior management and involving the end user but they do not give guidelines as to how this can be done. Systems developers do not really understand the social nature of systems development and find it difficult to adapt [Hirschheim & Newman, 1991]. The next section looks at some of the skills, besides the technical skills, that IS developers need to develop.

3.2 USER PARTICIPATION

It is important for all the stakeholders to have a say in the development of the IS. User participation is considered to be necessary for effective ISD and for systems to be accepted. User participation is called for in almost every IS development methodology according to Kirsch and Beath [1996]. The difference is in the degree to which users are able to influence that design [Damodaran, 1996]. The users are often seen as providers of information rather than decision-makers, for example.

There have been conflicting reports on user participation and its benefits to the IS

process. This section will look at different types of user participation and then take a look at how user participation has changed over the years. The methods that can be used to ensure effective user participation will then be described.

3.2.1 Levels of user participation

There is a difference between user participation and user involvement. User participation can be seen as the various behaviours and activities that the users or their representatives take part in during the process of ISD. User involvement is concerned with the psychological state of the individual and how they relate to the IS [McKeen & Guirmaraes, 1997; Kirsch & Beath, 1996]. User involvement is thus subjective and is determined by how relevant the system is to the person and the person's perception that their views were incorporated into the design of the system. With user involvement, the user must just be convinced that their views are represented, either by a colleague, a manager or someone else in whom they have confidence [Jones & Harrington, 1996].

Not all users can or want to actively participate in the development process. User participation, on the other hand, refers to the specific activities or behaviours that the users engage in during the design of the system.

Lawrence and Low [1993] determine three levels of user participation in systems development. These are:

- **Consultative participation** - In this type of participation the main decisions are made by information systems personnel with the user only acting in a consultive role.
- **Representative participation** - In representative participation a team is formed using representatives of the users and systems analysts and the team designs the system and manages the project.
- **Consensus participation** - This type of participation uses a democratic approach and tries to involve all users continuously throughout the design process. This is only feasible if there are only a few users.

The JAD technique, described in the next chapter can be used in both the representative and consensus levels of user participation. It helps bring users and developers together throughout the systems development cycle in order to improve communication and help users to reach consensus about what is needed in the system.

Dean, Lee, Pendergast, Hickey and Nunamaker [1998] suggest that different levels of user participation are needed throughout the Systems Development Life Cycle (SDLC). They suggest that user representatives, user groups and the user community as a whole will each play a part in the development process.

3.2.2 An historical perspective of user participation

Participation by users has long been acknowledged as important in ISD. This was especially true in Scandinavian countries where much research has been done in this area. There has, however, been some controversy as to what this participation should involve.

Clement and van den Besselaar [1993] did a study of papers on participative design over the years and have made the following observations:

- During the 1970's user participation was mostly concerned with providing users with knowledge about new technologies and how they would be used. They also helped users understand how their working conditions would be affected by the introduction of those new technologies.
- Trade unions were also involved in the 1970's but only so far as encouraging them to develop and implement their own technology solutions to control their activities.
- This changed during the 1980's when IT was being used more in offices and service industries and not only in the manufacturing arena. More women were using computers which led to greater involvement on their part in the design of computer systems.

- The emphasis during the 1980's was more involvement on the part of users and top management.
- During the 1990's trade unions did not seem to worry as much as they previously had about the use of technology and it became more difficult to get them committed to development projects. This is supported by Bjerknes and Bratteteig [1995] who say that trade unions are not as powerful as they used to be and that new methods of achieving democracy within an organisation have come to the fore.

Scandinavian researchers have been at the forefront of the field of participative ISD for many years. They have emphasized the importance of considering human requirements and the work activity of users when designing technical systems. This has led to a socio-technical approach where human-centric analysis is used to investigate the impact that a potential computer system will have on humans. It also considers ways in which technology can be designed more effectively for people [Sutcliffe, 2000].

It is important when designing a computer system for an organisation as a whole to realise that the ISD will need to be a compromise between various groups. The different interest groups or stakeholders will have partly conflicting goals and interests. This has led to a recognition that something more is needed than the socio-technical approach. The collective resource approach notes that developers must negotiate between workers and management or workers from different parts of the organisation in order to reach acceptable solutions for the organisation as a whole [Bjerknes & Bratteteig, 1995].

Business Process Reengineering (BPR) has been an important development in systems implementation during the 1990's. BPR involves a change process where it is not only the computer system which changes, but jobs may be lost, responsibilities change and employees are faced with new challenges. This process is one that is full of conflict and power struggles and means that IS developers need new skills to work in these situations [Bjerknes & Bratteteig, 1995]. Clement [1994] noted that there had been a move to the use of computers to empower the users. Users have been given the

information to enable them to do their jobs better. They no longer have to pass decisions to be made up to their superiors. This helps workers to be able to act more independently.

The people involved in this type of project need to be involved throughout the ISD in helping to define the system. Users and system developers need to develop more constructive and respectful relationships in order to foster a more democratic system [Clement, 1994]. The methods and rules used in JAD workshops try to foster this relationship as discussed in the next chapter.

3.2.3 Effective user participation

The choice of the user participants is important. Users should be chosen according to their role in the company, their experience, the respect that they receive from their peers, their interpersonal skills and their ability to pass knowledge on to other users. Users must be chosen across the spectrum of users from operational to management staff. Damodaran [1996] proposes that users should go on training in order to understand their role and ensure that they have the basic IT skills necessary.

One of the key strategies for successful user participation, according to Hunton and Beeler [1997] is that the user must want to participate. In order to facilitate this the project leader will need to provide the user with meaningful opportunities for participation.

Shared participation is better than token participation [Kirsch & Beath, 1996]. In token participation, the users play a minor role and in reality the IS personnel provide all the expertise and make all the decisions. Shared participation has the developers and the users working together towards a solution. Users are seen as the domain experts and IS personnel as the technical experts but they share roles, work together in a number of duties and coordinate with one another. Hunton and Beeler [1997] go a step further and suggest that the user should have control of the project. McKeen and Guirmaraes

[1997] have done research, which showed that user involvement must be high when the task or the system is complex.

Conflict can have a negative effect on information systems development. It should, however, be seen as a necessary and important part of the systems development process [Kirsch & Beath, 1996]. In the framework presented previously, argumentation is one of the orientations of systems development. IS developers must learn to resolve conflict in a positive way. Even user resistance is not necessarily bad, according to Hirschheim and Newman [1991]. Users are able to tell if something will work in their setting whereas system developers only look if it is technically feasible. They say that one needs to realise that there should be constructive conflict. When designers and users from different departments meet there may be conflict but the aim should be to reach consensus and methods of achieving this should be promoted. This would enable participation to be genuine rather than manipulative. They suggest creating encounters to reveal and resolve conflict. JAD workshops, which are described in Chapter 4, are one way of doing this.

Three principles for effective user-centred approaches are proposed by Nodek & McNeese, 1997]:

- Shared communication must be actively employed;
- knowledge must be able to be expressed without constraint; and
- the knowledge representation methods used must be compatible with the capabilities, limitations and needs of the stakeholders.

One of the problems of IS development has been the communication gap between the user and the IS developers. This can be exacerbated by the use of IS jargon and the insistence that users sign off specifications that they do not truly understand. Nosek and McNeese [1997] suggest that there are three factors that should be catered for in order to facilitate getting information from groups of users. They are:

- An active means of facilitating group communication must be employed.
- Users should be able to express their knowledge without constraint.

- The modelling methods used to model that knowledge must be understood by all the stakeholders.

This is supported by Checkland [1993] in his Soft Systems Methodology. Checkland says that there should be a relationship between the user and the developer and that the methods used must be suitable for both. These suggestions are also followed in the JAD sessions discussed in the following chapter.

Users should be chosen who feel that the new IS will be relevant and important to them [Hunton & Beeler, 1997]. They must also be representative of the users so that the other users will feel that they are involved. It is the perception of user involvement that is important to the success of the system [Lawrence & Low, 1993]. User representative will need to help the other users understand the objectives of the system, receive training and keep them up to date with the progress of the system. Champions must be sought who can lead from a users perspective. It is also important that the users come from the different departments that will be involved and that they have support from top management.

"*Central to the whole notion of user participation is the right of people to have a direct influence on matters that concern them in their work. It cannot be restructured simply to the design of information systems, but inevitably brings in wider elements of working life.*" [Clement & van den Besselaar, 1993, p.36]. If an organisation involves the users in the design process then they must be willing to introduce those designs otherwise the whole process will be merely an illusion [Bjerknes & Bratteteig, 1995]. The whole process must be self-sustaining in that even after the system has been rolled out, user participation is critical [Clement & van den Besselar, 1993].

The next chapter describes the process of JAD which is a method that tries to create a forum for effective user-developer interaction.

3.3 SKILLS NEEDED BY IS PROFESSIONALS

As IS development moves from the technical to the language and organisation domains and from the control to the sense-making and argumentation orientations, the skills needed by IS developers have changed. As Lee, Trauth and Farwell [1995, p.313] put it "*The requirements for IS professionals are becoming more demanding in multiple dimensions, particularly in the areas of business functional knowledge and interpersonal/management skills*".

In the sense-making orientation it is important for the IS developer to be able to find shared meanings with the users. Communication and interpersonal skills are imperative. Opportunities for developing shared meanings should be fostered. The IS developer must be able to act constructively within the sense-making situation. He or she should be able to know how to ask the right questions to get to the hidden meanings and taken-for-granted practices.

The argumentation orientation requires IS developers who can use rational argument and debate in order to achieve consensus. The ability to evaluate solutions and think critically is important in this orientation.

IS developers need to be prepared for working in an environment where user participation is the norm and where they will be expected to interact with people who are diverse from themselves. As users become more sophisticated in their use of personal computers, they will demand a greater say in the construction of their systems.

In a study done by Hunter [1993], users, sponsors, clients, systems analysts and business systems managers were asked to describe their experiences with good and bad systems analysts. They did the research at two organisations and found that communication skills were considered the most important factor with attitude holding second place. Communication skills were considered about twice as important as knowledge, which was in third place.

A survey, conducted by van Slyke, Kittner and Cheney [1998], found that general thinking, communication and interpersonal skills were considered to be the most important characteristics needed by IS graduates.

A more recent study by Doke and Williams [1999] determined what skills are considered important for seven different categories of IT jobs. Overall interpersonal skills came out first, with IS design and implementation second, IS analysis third, oral and written communication fourth and interpersonal behaviour and project management tied for fifth place. In seventh place was group dynamics. Once again, the soft skills were considered as important as the more technical skills.

Another way that some authors see the changing role of the IS professional is that the person becomes an agent for change and that they, therefore, need the skills to promote change within an organisation [Trauth, Farwell & Lee, 1993]. Effective change management requires IS professionals to work together with the users and prepare them for the changes. This does not mean just telling them what will happen, but rather means making sure all the diverse clients are served, listening to and valuing input from all the clients and sharing credit with those clients [Markus & Benjamin, 1996].

The skills of listening, negotiating, conflict management, persuasion and working in teams have thus become more important to the IS professional. Effective communication skills are considered essential to fostering teamwork [Spiegel, 1995].

3.4 IMPLICATIONS FOR IS CURRICULA AT TERTIARY INSTITUTIONS

IS departments at tertiary institutions have tended to prepare students very well for the control orientation of systems development. They have given the students the tools and techniques that they need to develop systems in a mechanistic way. There has been some neglect of the skills that the students need in order to be able to find shared meanings, practise argumentation and be effective in working with users, however.

"Employers want IS graduates who can think, communicate, and work well with others. They also want individuals who have a good grounding in basic IS skills such as systems analysis and design and database concepts. These employers seem to be less interested in more specific technology skills." [Van Slyke, Kittner & Cheney, 1998, p.10]

The environment in the tertiary institution tends to be one which is controlled and where there is only one best answer for a particular problem. In the real world, this is rarely true, however. Goyal [1995/1996, p.135] claims that "*Most students entering the job market struggle through the difficult transition from the university environment, which demands controlled, encapsulated thinking to the real world environment which demands creative, unstructured thinking*". Lecturers define problems and give "model solutions" to those problems instead of actively involving the student in the process.

The demand for IS professionals who are multi-dimensional across the fields of technology, business and interpersonal skills is increasing, according to a study done by Lee, Trauth and Farwell [1995] and all of these aspects should be built into an IS curriculum. As business needs to get new employees productive as soon as possible, the more of these skills that can be focussed on during the IS course, the better. This does not mean that the technical should be neglected, but that students should be able to augment their technical skills with the business and interpersonal skills [Todd, McKeen & Gallupe, 1995].

Another aspect that should be considered is that, in industry, IS professionals are expected to work in teams. Research has shown that people attracted to the field of IS often have a very low "*social need strength*" coupled with a high "*growth need strength*". This means that they do not feel the need to interact socially but are more achievement oriented and love a challenge [Spruell & Le Blanc, 1992]. Tertiary institutions often emphasize the individualistic rather than promoting group work among students [Moad, 1995]. Group work can give the students some of the essential skills of team work needed for their career. Communication skills like listening skills, the ability to manage conflict, the ability to deal with criticism assertively rather than aggressively, being able

to persuade and influence others, as well as how to negotiate are considered important in today's world and can be practised in the group environment [Rooff-Steffen, 1991].

The IS '97 Curriculum [Davis, Gorgone, Couger, Feinstein & Longenecker, 1997] recognises this need for the softer skills and has communication skills and interpersonal skills as two of the main characteristics needed by an IS graduate. This is shown in Table 3.2. Communications skills include listening skills, negotiating skills, interviewing skills, facilitation skills, observation skills and presentation skills. The skills of leadership, small group communication skills, small group organisation and working with diverse people are listed among the interpersonal skills needed.

Characteristic	With the ability to	Using the knowledge of .....
Communication	- accurately observe, note and explain observations of events - actively listen and express complex ideas in simple terminology - organise and make presentations - write memos, reports and documentation	- listening, observing and documenting - interviewing and speaking - negotiation and facilitation - presentation and interpretation of data - multimedia development and utilization - computer and video conferencing techniques
Interpersonal relationships	- effectively work with people of diverse backgrounds - effectively work with people at all corporate levels - lead and facilitate teams in a collaborative environment - develop win-win approaches - empathetically listen and seek synergistic solutions	- leadership, management and organizations - small group communications and motivation - organization, team and individual goal setting - shared vision and responsibility - cultural diversity

Table 3.2: Capabilities and knowledge expected for IS program graduates [Davis, Gorgone, Feinstein & Longnecker, 1997, p.12]

Personal and interpersonal skills are also deemed important in the Informatics Curriculum Framework 2000 (ICF-2000) for higher education [Mulder & van Weert, 2000]. This curriculum was developed by the International Federation for Information Processing (IFIP) for the United Nations Educational, Scientific and Cultural Organisation (UNESCO). They identify three broad categories of Informatics

professionals, namely information users, information appliers and information workers. While the IS'97 curriculum suggests that activities that foster communication skills and interpersonal skills should be integrated into the curriculum, the ICF-2000 curriculum goes further, and suggests that these skills should be credit bearing.

In their curriculum for Instrumental Users, for example, they suggest that 4 of the 20 credits should be dedicated to personal and interpersonal skills. (One credit is equal to one day of study.) For information workers, they suggest that at the final level 12 of the 160 credits should be on the topic of personal and interpersonal skills. This is added to the 17 credits of the lower levels to give a total of 29 credits, or 29 days of study on the topic.

Fostering these soft skills in IS students, while still finding the time for them to learn the technical skills is a difficult process.

3.5 CONCLUSION

This chapter answered the research questions: "*What is involved in IS development?*", "*What social skills are needed by an IS developer?*" and "*Why should tertiary insitutions help IS students develop interpersonal and group skills?*".

The world of ISD is changing and as it does the skills needed by IS professionals are changing. It is no longer sufficient to have only technical skills. These need to be augmented with business and interpersonal skills. As the IS developer is asked to move from the control orientation, where all aspects of the system were seen as being able to be predefined, to the sense-making and argumentation orientations, his or her skills in communication, negotiation, achieving consensus and debating must improve.

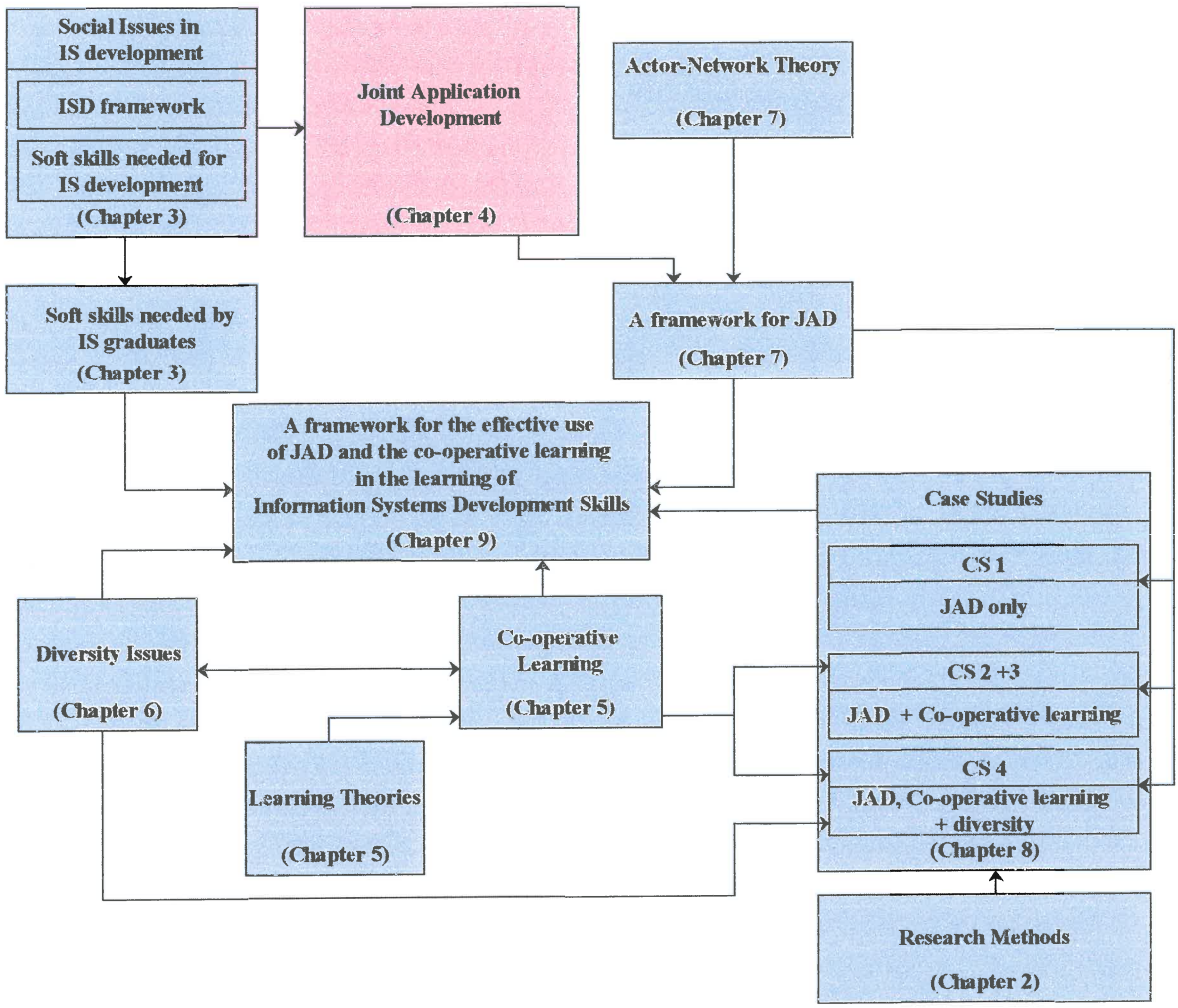
It is these skills, together with the more technical ones, that make a successful IS developer, and it is these skills, together with the more technical ones, that need to be

developed in students of IS. This thesis looks at how JAD can help students to develop some of these skills while helping them to learn the modelling tools.

JAD is a method of bringing together the different user groups and IT developers in order to facilitate understanding of the system to be built. The next chapter describes how this technique is used in industry.

Chapter 4

Joint Application Development



Chapter 4

Joint Application Development

Good communication among the systems developers, users and top management is essential for the production of a satisfactory system. Lockwood [1989] poses the following questions: "*How can systems professionals deal with the nagging problem of getting top management and users to pay attention to systems details during the crucial early phase of requirements specification? Furthermore, how can honest disagreements in requirements specifications be resolved in a timely manner and with a consensus of the people involved?*" JAD offers one method of trying to answer these questions.

In terms of Hirschheim et al.'s [1996] framework, JAD can be seen as a method of trying to achieve shared understanding and manage the process of argumentation in order to support the organisation and communication domains. JAD focusses on "*facilitated interactions between users and designers wherein group techniques are employed for eliciting and refining ideas.*" [Carmel, Whitaker & George, 1993, p.40]

Eliciting information from users in the traditional way has meant relying on many user interviews and surveys. Serial interviews with a large number of users is inefficient. It is also difficult to resolve conflicts in requirements between the different user groups [Dean, Lee, Pendergast, Hickey & Nunamaker, 1998]. JAD is a method that tries to enable all the stakeholders to reach consensus on requirements for a proposed system. Central to JAD is the structured workshop. During the structured workshop, a carefully selected group of people from the users and developers, gather to work towards a common goal or set of goals [Geier, 1996]. The workshop can be anything from a couple of hours to days long.

JAD is based on four philosophical principles [Sims, 1998, p.1]:

- The people who do the job have the best understanding of that job.

- People who are trained in IT have the best understanding of the possibilities of that technology.
- Information systems and business processes rarely exist in isolation. They transcend the confines of a single system and work across related departments. The people working in those different departments have valuable insight on the role of the system in the larger community.
- The best information systems are designed when all of these groups work together on a project as equal partners.

Some people define JAD as Joint Application Design and only look at the design of the project. JAD can, however, be used throughout the systems development life cycle (SDLC) and it is for this reason that it is termed Joint Application Development in this thesis. The diagram in Figure 4.1 shows how Gottesdiener [1994] of EBG Consulting sees methods of using JAD during the different life cycle phases.

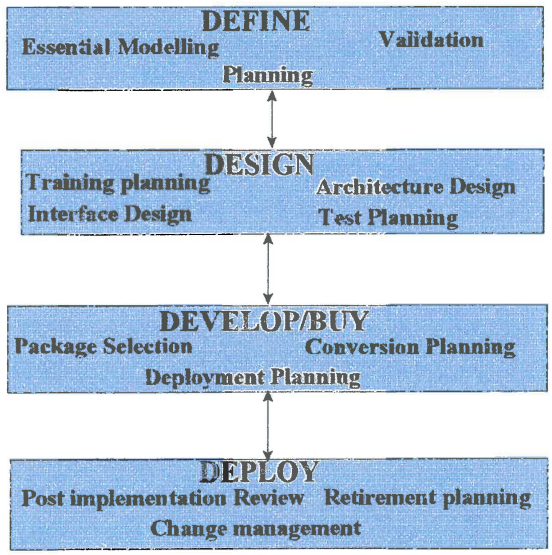


Figure 4.1: Uses for JAD in the Systems Development Life Cycle [Adapted from Gottesdiener, 1994]

JAD meetings early in the SDLC deal with high-level issues like defining the objectives and the scope of the system or decomposing the domain into smaller parts. Increased detail is required as the later design phases are reached [Carmel, Whitaker & George,

1993].

The theoretical basis for JAD is minimal. Many of the ideas for the running of the JAD workshops comes from the study of group dynamics. Carmel, Whitaker and George [1993] note in their article in Communications of the ACM, that there has been little academic interest in JAD. Most of the literature on JAD thus comes from practitioner journals rather than high-quality research journals.

This chapter will first give some detail on the composition of the team for the JAD workshop. The JAD process will then be described. Problems and techniques for promoting effective participation will be presented within the description of the various JAD processes. Lastly some perceptions on the value of the JAD process will be discussed.

4.1 THE JAD PARTICIPANTS

It is important to make sure that the right people attend the JAD sessions. There should be participants from both the Business and Technical sides. Potential users from the Business side are included to give their input on how the system should be designed or implemented. The developers are present to analyse the needs, as expressed by the users, and to gain clarity on what the system should do. A JAD facilitator is used to control the meeting and act as mediator or guide and a scribe records the proceedings of the meeting. Each of these will be described in more detail below.

It is suggested that there be less than 15 people in the JAD workshop [Lockwood, 1989; Knowles, 1995]. A ratio of 3:1 of business users to technical personnel is considered to be good. The roles of the different participants and guidelines for choosing them are given below.

4.1.1 The users

It is important for the success of a project that there is an executive sponsor. This person should try to attend at least the first JAD session and should be available throughout the period of the JAD sessions. The sponsor is usually from the End user community or a Vice President of the company [Netmation, 1998]. The sponsor ensures that the users and technical staff are given the time and the financial support needed to develop the system and attend JAD sessions.

End user involvement is necessary for JAD to succeed. It is the users who give the input into the meeting. One of the problems with user involvement has been with IS developers not listening and not giving opportunities to users to participate effectively. JAD tries to overcome this by getting users together to define or test a system. The role of the user is expanded and they collectively are asked to articulate, negotiate and help develop the system [Purvis & Sambamurthy, 1997]. The users present their differences in their expectations for the new system and negotiate the differences within the structure of the JAD workshop.

In order to achieve this, users must be chosen who know the business. It is important to have a mix of the decision makers from a department and the operational staff who know about the day-to-day operation of the department [Knowles, 1995]. Lockwood [1989] suggests that the users in the JAD workshop should be made up of 10% executives, 20% managers and 70% operational staff. If the system is to serve more than one department then the users must be carefully chosen to represent the different departments. Problems can occur if a critical person is forgotten [Wood & Silver, 1995].

The users should also be able to communicate effectively in order to describe their needs, problems and processes to the others in the workshop [Knowles, 1995]. They should be committed to the objectives of the workshop, for example, designing a quality system. Users should be involved, not only at the requirements stages, but throughout the development of the project. They should know that they will be required to work in

the JAD sessions, follow up on those sessions, evaluate prototypes and even beta test the systems. This participation will enable users to feel that they are part of the project rather than just having someone else's ideas forced upon them [Dodson, 1994].

4.1.2 The IT specialists

The IT specialists' primary responsibilities are to advise the users and to listen and make sure that they get enough detail to be able to build the system.

The IT specialists should be people who understand the organisation and the business area involved. They should be good listeners and should be able to empathise with the end users [Netmation, 1998]. Some authors suggest that they should be silent and only observe the proceedings [Botkin, 1994; Lockwood, 1989]. The IT personnel should definitely not be allowed to take over and control the meeting [Geier, 1996]. They are there to learn rather than to get the users to rubber stamp decisions that they have made previously.

Carmel, Whitaker and George [1993] suggest that historically JAD required silent participation from IS members but that JAD workshops now emphasize the idea of the JAD group being a team. This is supported by Jackson and Embly [1996] who also see users and IT personnel working as a team in the JAD sessions.

4.1.3 The facilitator

The facilitator is key to the JAD workshop. The facilitator is used to guide the team towards the goals set for the workshop. The facilitator is also the person responsible for the planning of the workshop and making sure that the documentation is prepared after the workshop. The facilitator's role in the workshop is to help guide the discussion in order to achieve the goals of the workshop. This involves keeping the group on track and making sure that the participants understand one another. Grove Consultants [quoted in EBG Consulting,1998, p.1] say that facilitation is "*the art of leading people*

through processes toward agreed-upon objectives in a manner that encourages participation, ownership and productivity from all involved".

The facilitator is not there to offer opinions and act as a consultant. His or her job is rather to keep the team members actively engaged [Geier, 1996]. Rettig [1990] suggests that the facilitator is there to get the best possible use of the resources of the group. The facilitator must also make sure that the team members do not get expectations which are too high. Limitations due to budget, technology and time table should be explained to users by the facilitator [Dodson, 1995].

The facilitator will need to know how to handle people and get the best out of them. He or she will also need to be respected by the other team members [Netmation, 1998]. A facilitator must be able to [Bacal, 1998; Wood & Silver, 1995]:

- remain objective;
- be skilled at understanding and handling group dynamics;
- adapt to changing situations;
- think quickly and logically;
- use time and space intentionally;
- evoke participation and creativity from others;
- ask for the opinions of others rather than always offering their own;
- lead the group to consensus rather than compromise;
- demonstrate professionalism, self-confidence and authenticity;
- listen without interrupting;
- communicate clearly and expressively;
- build relationships rather than be task oriented;
- be more like a coach than a scientist and more like a counsellor than a sergeant; and
- keep the big picture in mind while working on the detail.

JAD facilitation skills may need to be learnt as many of these skills will not come naturally to someone. Training in group dynamics is essential for a JAD facilitator

[Davidson, 1999].

Some authors suggest that it may be worth considering an independent consultant as a facilitator in order to assure neutrality and avoid having the person perceived as being biassed [Geier, 1996; Lockwood, 1989]. Wood and Silver [1995] maintain that the person should be politically neutral. This does not seem to be common in practice, however. As Knowles [1995] reports, their research has shown that 89% of facilitators were from the IT departments of their organisations.

4.1.4 The scribe

The scribe is the person who records the proceedings of the JAD workshop. In order to do this the person will need to be more than a secretary as they will need to have some knowledge of the modelling techniques and the subject matter [Geier, 1996]. The scribe will need to learn to capture important decisions made, who made them and why they were made. This documentation forms an important record of the session which can be used for later reference.

The scribe works closely with the JAD facilitator to make sure that all decisions are captured. All participants should be able to ask the scribe to ensure that a particular point has been documented [Netmation, 1998]. The scribe should also act as a sounding board and ask for clarity on any point that is not clear [Sims, 1998]. It may be necessary for the scribe to use CASE tools in order to capture information. The person should, in that case, be an expert in the tool as they will need to input the model as the discussion progresses. The task of the scribe is not an easy one and should not be underestimated.

4.2 THE JAD PROCESS

In order for the JAD process to be successful, the workshops must be properly planned, executed, documented and acted upon. These steps in the JAD process will be

described briefly below.

4.2.1 Preparation for the workshop

JAD workshops must be properly planned in order to achieve their goal. Some authors divide the preparation into three steps, namely project definition, research and preparation [Wood & Silver, 1995; Damian, Hong, Li & Pan, 1999].

During the project definition phase, the purpose, objectives and scope of the session must be determined and agreed upon. The participants should be carefully chosen as specified in the previous section. Each participant should be scheduled to attend and should understand why they need to be there. They should thus all be talked to before the JAD workshop [Geier, 1996]. This will increase their commitment to the project. The JAD facilitator and the scribe must be assigned. Sometimes the job of informing the participants of their role is assigned to the facilitator. Management commitment must be gained for the project and for the time that the participants will need to spend in the workshop [Gottesdiener, 1994].

Research is needed to determine what needs to be placed on the agenda and to determine how work is presently being done [Wood & Silver, 1995]. Some of the user requirements are explored. This research is then used to set up an agenda and a time frame for the workshop [Gottesdiener, 1994]. Participants should be told of any reading or other work that they may need to do before the workshop.

The preparation phase involves preparing everything that is needed for the JAD workshop. The venue for the JAD session should be chosen. It is usually best if the location can be away from the work environment of the participants so that they do not get called out to deal with problems during the workshop [Wood & Silver, 1995]. The actual room where the JAD session will be held is usually set out as indicated in Figure 4.2 [Carmel, Whitaker & George, 1993].

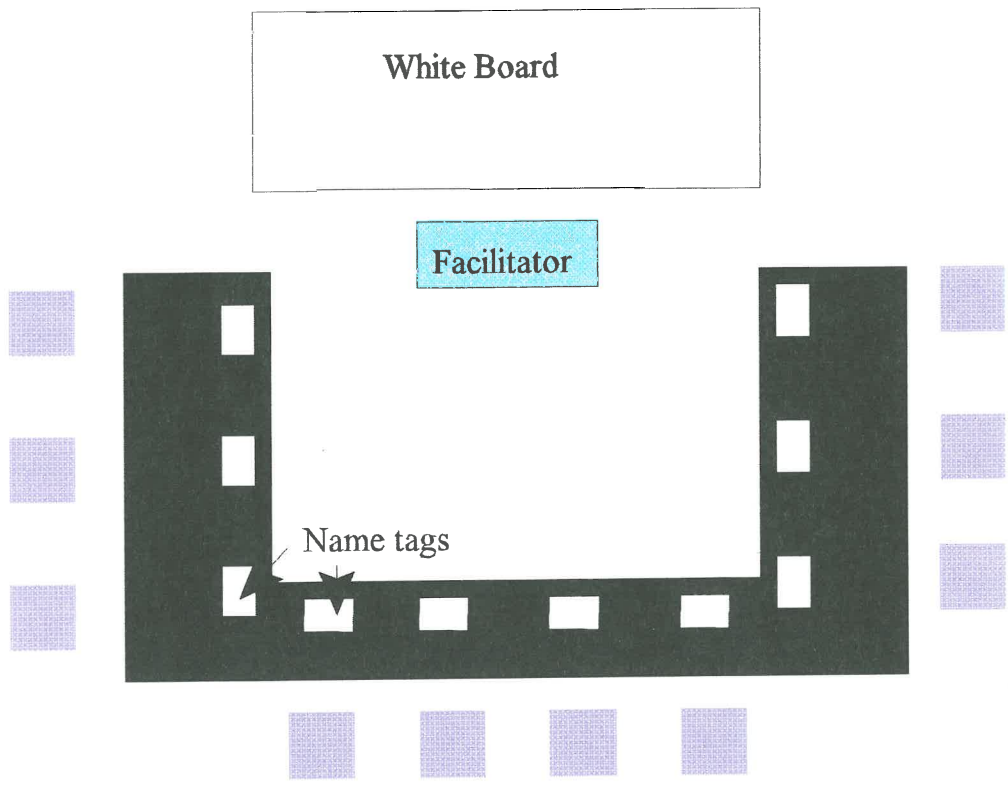


Figure 4.2: A typical JAD room

The participants sit in a horseshoe shape so that they can all see each other and the board. Visual aids might be prepared and these might be put onto the walls around the venue. Data projection or overhead projectors might also be used and should be planned for. Logistics like food, transport and, if necessary, accommodation must also be arranged.

4.2.2 The structured workshop

The workshop should be started in a positive way. Make sure that everyone understands the objectives for the workshop. The ground rules for the session should be established first. Some of these ground rules might be [Gottesdiener, 1994; Geier, 1996; Handley, 1998]:

- All participants are equal. Managers should be prevented from dominating the session, especially when the operational detail is being determined.
- Only one conversation will be allowed at once.
- No idea is bad - everyone should respect the ideas of the other group members.
- All speakers will be allowed to finish their thought without interruption. Everyone in the group will become active listeners and try to listen without letting their preconceived ideas interfere with their ability to hear.
- The group must accept responsibility for the deliverables.
- Off the target discussions will be limited but a record will be kept of any issues that should be returned to at a later date.
- Everyone must be on time for each session of the workshop.
- Computer jargon should be avoided.

4.2.2.1 Working towards a common goal

The JAD facilitator should control the workshop. He or she should make sure that participants stick to the agenda and work towards the agreed-upon goal. The facilitator should not dominate the session him- or herself as it is the opinions of the users that are sought, and not those of the facilitator [Geier, 1996]. JAD is a time-consuming activity as it requires a number of key personnel to be available for an extended period of time. The facilitator must, therefore, make sure that this time is used effectively.

4.2.2.2 Communication

The facilitator stands at the board at the front of the U of the horseshoe, in order to write down decisions on the board while controlling the meeting. The users and the developers should not be placed on opposite sides of the horseshoe as this will create the impression of "us" and "them". Separating participants will also help to reduce the opportunities for people to carry on their own private conversations during the session.

IT personnel should try to be clear when asking questions and discussing with users.

They should try to avoid computer jargon. While users may pretend to understand what they are talking about, it may be discovered later that they were covering for their ignorance [Wood & Silver, 1995].

Simple diagramming techniques should be utilized in order to express the requirements. Whatever technique is chosen, it should be one that is understood by both the users and the developers [Geier, 1996]. The scribe should record the model as well as any side issues for discussion later or detail that should be remembered. Users can get frustrated with IT staff if they use models as a form of "computer jargon" to confuse them [Davidson, 1999].

Some of the JAD team members might be shy or withdrawn while others might be dominating. This can cause problems for the facilitator who should be trained to handle this. The problem is even more serious if the dominator is the boss of some of the other team members or is the person responsible for the system. The dominating member may be spoken to by the facilitator during a break. The idea behind JAD should be explained to the person. It is even better to explain the concepts before the time so that the problem does not occur [Wood & Silver, 1995]. The shy person may be asked direct questions to try to foster his or her participation. While asking them the questions, the facilitator may also have to try to stop the other people from answering them.

Another problem occurs when employees know their jobs well but have never before been asked to communicate what they do to someone else. They may have difficulty doing this in a way that is clear to the rest of the group. Again it is the job of the facilitator to be patient and supportive to the group member and to persevere until the team has a clear picture of the person's work [Dodson, 1994]. In order to do this the facilitator should have done his or her homework so that they themselves have enough of an understanding of the system to ask the right questions of the person.

Listening to the other people in the team is important in a JAD session and this must be stressed with the participants. Horowitz [1996] suggests that poor listening can add to

the cost of systems development. IS people are notorious for poor listening. They tend to race ahead and try solving the problem before they truly understand what is needed. The facilitator can use the method of reflexive listening where he or she paraphrases what the person has said in order to make sure that the person is understood by all.

4.2.2.3 Group dynamics

Group dynamics techniques are needed for inspiring creativity, solving disagreements and handling communication within the group [Carmel, Whitaker & George, 1993].

Conflict is an important part of the design process. It can be educational if it is handled correctly [Purvis & Sambamurthy, 1997]. Gottesdiener [1994] claims that "*healthy disagreement can be a source of creativity and strength to the whole group and should be encouraged.*" The success of the system could be related to the facilitator's ability to manage conflict and achieve consensus. The potential for conflict is especially high in information systems that cross departmental boundaries, especially if political issues mean that there is already tension between those departments. The conflicts should not be ignored in order to get a workable system quickly as this may lead to a less useful system. Purvis and Sambamurthy [1997] contend that the diverse perspectives of the team members must not be stifled, their commitment must be maintained and group cohesiveness should be encouraged while solving conflicts. This is not an easy task and is another aspect of facilitation that may need to be taught to the facilitator.

Compromise is not consensus according to Constantine [1992]. Compromise tries to find a middle road which may leave you with a solution that is worse than any of the original alternatives. Consensus tries to take advantage of all the skills and experiences of the members of the JAD team. Constantine gives some guidelines that can help with this. He says that each of the team members should be persuaded that it is possible to reach consensus and that it is more important to get the best design than it is to get their preconceived idea into the result. Each idea must be judged on its own merits and should not be seen a part of a point-scoring system where concessions made in one area

are traded for concessions in another area. The facilitator is meant to guide this process [Rettig, 1990].

Another skill that needs to be learnt by the team and fostered by the facilitator is the skill of separating fact from opinion. The team must be able to get reliable information. Opinions are not bad and are often useful but they should not be confused with facts and true analysis [Constantine, 1992].

Group cohesion concerns the positive attitudes that the team members have towards the group. It is the individual's sense of belonging to that group and his or her feelings associated with being a member of that group. Group cohesion should be strengthened as this makes the group more effective according to Jones and Harrison [1996]. They found in their research that higher levels of perceived group cohesion led to increased perceived IS team performance.

One problem with groups is that positive factors like group cohesion can force members to be compliant with group positions and this can lead to the problem of group think. This is where the group agrees because they feel that they should agree rather than because they actually think that the decision is right [Gottesdiener, 1994]. One method of overcoming this is to view the topic from another perspective in order to get new ideas.

4.2.2.4 Role of the facilitator

As can be seen from the above, the facilitator needs to be experienced in modelling techniques, group dynamics, conflict management and other interpersonal skills in order to achieve the best results from the workshop. A competent facilitator is essential to the workshop but he or she cannot work alone. The scribe, the users and the IT professionals must all be committed to the process and must be willing to learn the skills necessary for promoting trust within the team.

4.2.3 Post workshop

All documentation concerning the meeting should be published and distributed as soon as possible after the meeting. This will enable the participants to check what has been done and provide corrections to the scribe [Botkin, 1994]. The checking should be done while the JAD workshop is still fresh in the participants' minds [Wood & Silver, 1995]. Any corrections are consolidated into the document which is then redistributed to the group until it is correct.

It is useful if a CASE tool has been used for the documentation as this can then serve as a repository of the decisions taken. The repository, whether it be in a CASE tool or some other form of documentation, will become the group memory for use during the development of the system.

4.3 PERCEPTIONS AND RESEARCH OF JAD

Many claims are made about the effectiveness of JAD. Some of these claims are [Gottesdiener, 1994; Purvis & Sambamurthy, 1997; Carmel, Whitaker & George, 1993]:

- JAD enables IT people to learn about the organisation while enabling users to learn about technology.
- Communication is improved among users, designers and other parties.
- A better understanding of the requirements is achieved, thus realising a better quality product.
- The creeping scope problem is reduced.
- Productivity is increased.
- The users sense of commitment and ownership is increased as their control over the project is promoted using the JAD workshops.
- Cooperation, understanding and teamwork are promoted.
- Consensus is sought and managed more effectively.
- Users resolve their differences in the workshop rather than having the IT personnel try to resolve any conflicts.

- The users are more satisfied with the system.

Few of these claims have been researched, however. As Carmel, Whitaker and George (1993) say, most of the ideas for the effectiveness of JAD come from the study of group dynamics, but these ideas have not really been researched much in the IT sphere. Only two research studies on the effectiveness of JAD could be found in recent literature.

The first research study compares the effects of JAD and traditional design methodology with regard to their perceptions of the design success [Purvis & Sambamurthy, 1997]. They looked at three perceptions of users and designers, namely:

- Perceptions with respect to the user-designer interactions;
- perceptions with respect to the effectiveness of consensus management; and
- perceptions with respect to user acceptance of the designs.

They found that there was significant indication that both users and developers agreed that the JAD methodology promoted better interactions among the participants. This included more user participation and influence, greater partnerships and improved communication. The developers thought that JAD was superior in promoting effective consensus management and user acceptance. The users did not have this perception, however. There was no significant difference in the users perceptions of consensus management and user acceptance between the JAD and traditional methodologies.

In trying to explain this, the authors suggest that the problem may be due to a number of factors. The first is that the developers acknowledged that they were not very confident in using the JAD method and had less experience with the method. The developers' facilitation skills were also not as good as they should have been and this may have been detrimental to the process. Another factor that they identified was that the developers were glad to pass some of the responsibility for the design onto the users. JAD shortened the life cycle for them and allowed them to resolve conflicts more efficiently within the JAD workshop. The users, however, did not like to confront the differences in their expectations and needs in the JAD session. They had to reach

consensus with other members themselves instead of leaving this task to the IT personnel. With inexperienced JAD facilitators the problems in reaching consensus were exacerbated. The users had to give more commitment to the system than with traditional methodologies.

Purvis and Sambamurthy [1997] conclude by saying that it is important for the success of the JAD process to gain the same degree of enthusiasm from users as is expressed by the developers.

The second research study was reported by Davidson [1999]. He studied three organisations that were very competent in the use of JAD in order to determine whether JAD was perceived to be helpful in improving systems.

He found that in most of the projects, people felt that the analytical tasks and models had been well-documented in the JAD workshops. The specifications from the JAD sessions were partially or completely applied during the next stage in the project in 70% of the projects. Looking more closely at this result, he found that while the specifications were used in 100% of the small projects and 67% of the medium size projects, they were only used in 40% of the more complex projects. It seemed like the size and complexity of the project influenced the effectiveness of JAD as a systems specification method.

Most of the people who participated in the workshops found that they were well conducted and were supportive of the method. Some of the benefits that they mentioned were: better quality of requirements were defined, better relationships between the IS team and the users and the chance to learn about how the business worked. The IS developers felt that it was a more efficient use of their time. Many of these factors were mentioned as advantages of JAD in the non-researched list at the start of this section.

Davidson [1999] also found that the JAD workshops adapted to suit the organisations within which they were run. Most of the organisations did not feel that the users could participate full time in JAD workshops so they would have multiple shorter JAD sessions.

The JAD participants were also often the managers rather than a mixture of management and operational staff. These factors limited the potential effectiveness of JAD according to Davidson [1999]. Another problem was the use of IS models of which the user participants had no prior knowledge. This led to communication difficulties and frustrations on the part of the users.

It would seem from this research that JAD is more useful in the smaller to medium-size projects. How JAD is actually practised compared to how it is theoretically defined can also influence the effectiveness of JAD in the systems development process.

4.4 JAD IN THE THEORETICAL FRAMEWORK FOR ISD

JAD can be used as a tool within the sense-making and argumentation orientations of Hirschheim et al's [1996] framework discussed in the previous chapter.

4.4.1 The use of JAD in the sense-making orientation

The sense-making orientation tries to get a shared vision among the different players as to what is needed in an information system. This involves building shared meanings and shared use of language between developers and users and implies the direct participation of both. Formal and informal interaction is required among the different role players.

The JAD workshop can be used as a tool to help to get this shared vision and promote interaction between the different role players. The objectives of the JAD session include making sense of the needs of the organisation and the needs of the various users represented in the workshop. Sense making is thus very much a part of the JAD workshop.

4.4.2 The use of JAD in the argumentation orientation

The argumentation orientation aims to achieve clarification and required developers to justify their claims and provide reasons for what they are doing. Rational argument is required where everyone has a chance to put forward his or her ideas and debate is fostered. Each solution is investigated critically and debated until consensus is reached.

This is what should happen in the JAD session. The JAD workshop provides a structured forum for debate, providing evidence and reaching consensus between the different players in the IS development team. Getting the different users and IT developers together helps the team to challenge established traditions and its own thinking in order to promote change if necessary. The JAD workshop can, therefore, be used very effectively as a tool to promote an argumentative development orientation.

4.5 CONCLUSION

Chapter 4 answered the research questions: "*What is JAD?*", "*Why do people use JAD in industry?*" and "*How does JAD work in industry according to the literature?*". JAD sessions are a method of bringing together people with different backgrounds into one structured workshop in order to work towards a common goal - developing a system that will suit all the players. The people in the workshops communicate, listen to one another, negotiate and solve conflicts while working in a small to medium size group. The facilitator is required to show leadership, listen, help resolve conflict situations and keep the members working towards a goal. Sense-making and argumentation skills are fostered.

These are all skills that IS students should learn. In addition they need to learn the modelling techniques that would be required to design a system. This study looks at how the techniques of JAD can be brought into the classroom and combined with the techniques of co-operative learning in order to learn effectively. The next chapter looks at co-operative learning and how it can be used to promote learning in the classroom.