17

Information systems development methodologies: a broader perspective

D E Avison
Department of Management, University of Southampton
Southampton SO17 1BJ, UK, tel: 44 1703 592563, fax: 44 1703 593844
e-mail: dea@socsci.soton.ac.uk

Abstract

This paper first provides a historical perspective on approaches to developing information systems and argues that there are major weaknesses associated with the conventional waterfall model and the methodologies which followed. The paper suggests that a contingency approach to information systems development has much to offer and looks at Multiview, which is described as an exploration in information systems development. Some strengths and weaknesses of this contingency approach are highlighted and a new version of Multiview offered. This description enables a further discussion of information systems development and suggests that human and organisational aspect are at least as important as the technical ones which tend to be emphasised. Information systems development is seen as first a social process, though it will contain technical aspects. This social process is examined in more detail illustrating the arguments, for example, with different views of the systems analyst and the problem situation in this process. Such a broad approach also suggests that the area of which information systems development is a part, is multi-disciplinary where technology and computing are by no means dominant.

Keywords

Action research, contingency, Multiview, social process, soft systems, information systems development

1 AN HISTORICAL PERSPECTIVE

1.1 Pre-methodology era

Early computer applications were implemented without an explicit information systems development methodology. The emphasis of computer applications development was on programming, with systems developers technically trained but rarely good communicators,

S. Brinkkemper et al. (eds.), *Method Engineering*

nor were they systems analysts. The needs of the users were rarely well established with the consequence that the design was frequently inappropriate to the application. The dominant 'methodology' was rule-of-thumb and experience. This led to poor control and management of the project. Most emphasis was placed on maintaining present systems to get them right rather than developing new ones. Management were not getting value for money, and there was a growing appreciation of the potential role of the systems analyst and the need for a methodology to develop information systems.

1.2 The waterfall model

The life cycle or waterfall model (for example, Daniels & Yeates, 1971) of feasibility study, systems investigation, analysis, design, and implementation, followed by review and maintenance, became that methodology. It was widely used in the 1970s and is the basis for many methodologies that followed. It is well tried and tested. The feasibility study attempts to assess the costs and benefits of alternative proposals enabling management to make informed choices. The use of documentation standards helps to ensure that proposals are complete and that they are communicated to users and computing staff. The approach also ensures that users are trained to use the system. There are controls and these, along with the division of the project into phases of manageable tasks, help to avoid missed cutover dates. Unexpectedly high costs and lower benefits are also less likely.

However, there are serious limitations to the approach along with limitations in the way it is used. Some potential traps are (Avison & Fitzgerald, 1995):

- Failure to meet the needs of management (due to the concentration on single applications, particularly at the operational level of the organisation)
- Unambitious systems design (due to the emphasis on the existing system as a basis for the new computer system)
- Instability (due to the modelling of processes which are unstable because businesses and their environments change frequently)
- Inflexibility (due to the output-driven orientation of the design processes which makes changes in design costly)
- User dissatisfaction (due to problems with the computer-orientated documentation and the inability for users to 'see' the system before it is operational)
- Problems with documentation (due to its computer rather than user orientation and the fact that it is rarely kept up-to-date)
- Application backlog (due to the maintenance workload as attempts are made to change the system in order to reflect user needs).

As an answer to these criticisms, there have been a number of movements. The first is to reject methodologies by either playing lip-service to their use or fail to do even that. The second is to improve the traditional waterfall model by the inclusion of techniques and tools along with improved training so as to reduce the potential impact of these problems. A third movement is the proposal of new methodology themes and methodologies which are very different to the traditional waterfall model. A fourth movement is to suggest a more flexible contingency approach to information systems development reflecting the different problem situations that occur. We will look at each of these in turn.

1.3 Rejecting the methodology approach

One reaction to the unsatisfactory use of methodologies is the overt decision not to use any methodology when developing information systems (or the covert decision to pay only lip-service to them). A chosen methodology may not have been appropriate for the organisation and there has been a backlash against formalised methodologies. Their use has not always led to productivity gains. Methodologies have also been criticised for being over complex, for requiring significant people skills and expensive tools, and being inflexible and difficult to adopt.

More fundamentally, it has frequently been found that the existence of a methodology standard in an organisation leads to its unthinking implementation and to a focus on following its procedures to the exclusion of the real needs of the project being developed. In other words, the methodology obscures the important issues. De Grace and Stahl (1993) have termed this 'goal displacement' and talk about the severe problem of 'slavish adherence to the methodology'.

Wastell (1996) talks about the 'fetish of technique' which inhibits creative thinking. He takes this further and suggests that the application of a methodology in this way is the functioning of methodology 'as a highly sophisticated social device for containing the acute and potentially overwhelming pressures of systems development'. He is suggesting that systems development is such a difficult and stressful process, that developers often take refuge in the intense application of the methodology in all its detail as a way of dealing with these difficulties. Developers can be seen to be working hard and diligently, but this is in reality goal displacement activity because they are avoiding the real problems of effectively developing the required system. Users, analysts and managers thus find that the great hopes of some in the 1980s that methodologies would solve most of the problems of information systems development have not come to pass and this has led some to reject methodologies completely and others to use them as a social defence only.

1.4 Improvements to the waterfall model

Since the 1970s, there have been a number of developments in techniques and tools and many of these have been incorporated in the methodologies exemplifying the modern version of the waterfall model. Techniques incorporated include entity-relationship modelling, normalisation, data flow diagramming and entity life cycles. Tools include project management software, data dictionary software, drawing tools and computer-assisted software engineering (CASE) tools. The incorporation of these developments address some of the criticisms discussed in section 1.2, but give grounds to the potential criticisms of Wastell (1996). The data modelling techniques suggest that the waterfall models now are more balanced between process and data modelling rather than having a purely process modelling emphasis. The documentation has improved, thanks to the use of drawing and CASE tools, and it is more likely to be kept up to date and be more understandable by non-computer people. Further, CASE tools can be used to develop prototypes which enable users to assess the proposed information system in a far more tangible way and can speed up delivery of the operational system. The blended methodologies SSADM (Eva, 1994) and Merise (Quang & Chartier-Kastler, 1991) could be said to be updated versions of the waterfall model, and this updated waterfall model is the basis of many modern student texts and courses in information systems. Although these improvements have brought the basic model up to date, many users have argued that the inflexibility of the life cycle remains and inhibits most effective use of computer information systems.

1.5 New methodology themes and methodologies

Over the last ten or fifteen years, there have been many methodologies, some of which are as structured as the waterfall model, but reflect different movements in information systems development. They include incorporating ideas from systems thinking, typified by soft systems methodology (SSM) (Checkland, 1981 and Checkland & Scholes, 1990) which addresses the needs of management and the organisation as a whole; considering strategic issues, such as critical success factors (Bullen & Rockart, 1984), again, addressing the needs of management; business process re-engineering (Hammer & Champy, 1993), looking more fundamentally at the way the organisation does things (traditional information systems development is often accused of merely computerising present ways of doing things rather than improving things more fundamentally); object-orientation (Booch, 1991 and Coad & Yourdon, 1991) which unifies many aspects of the information systems development process and thus avoids the difficult combination of process and data approaches in one methodology; participation, such as ETHICS (Mumford, 1995) and joint requirements planning (JRP) and joint applications design (JAD) (Martin, 1991), where major consideration is given to the role of users and other stakeholders in the information systems development process, indeed, where the users rather than the technologists drive the process and therefore address the problem of user dissatisfaction that was inherent in traditional systems analysis; and the related emancipatory approaches where systems are developed which permit emancipation through rational discourse, typified by the UTOPIA project (Bodker *et al.*, 1987). All these approaches address some of the weaknesses of the traditional waterfall model and have been adopted by organisations. However, many users find some of them either unnecessarily complicated, expensive (in skills required and tools used) and difficult to adopt or, if this is not the case, narrow in their applicability and scope.

1.6 Contingency approaches to information systems development

Many users of methodologies have found the waterfall model and the alternative methodologies unsatisfactory. Most methodologies are designed for situations which follow a stated or unstated 'ideal type'. The methodology provides a step-by-step prescription for addressing this ideal type. However, situations are all different and there is no such thing as an 'ideal type' in reality. Situations differ depending on, for example, their complexity and structuredness, type and rate of change in the organisation, the numbers of users affected, their skills, and those of the analysts. Further, most methodology users expect to follow a step-by-step, top-down approach to information systems development where they carry out a series of iterations through to project implementation. In reality, in any one project, this is rarely the case, as some phases might be omitted, others carried out in a different sequence, and yet others developed further than espoused by the methodology authors. Similarly, particular techniques and tools may be used differently or not used at all in different circumstances.

A contingency approach is therefore suggested as a more realistic and useful methodology. Multiview (Avison & Wood-Harper, 1990) is an example of such an approach and this paper looks at this in more detail in the next section. It is a contingency framework in that it will be adapted according to the particular situation in the organisation. The authors are concerned to show that information systems development theories should be contingent rather than prescriptive, because the skills of different analysts and the situations in which they are constrained to work always has to be taken into account in any project. Each application of Multiview forms a new and original methodology. There are potential problems of the contingent approach and in examining Multiview, these potential criticisms ought to be considered. First, some of the benefits of standardisation might be lost. Second, there is a wide range of different skills that are required to handle many approaches. Third, the selection of approach requires experience and skills to make the best judgements. Fourth, they implicitly or explicitly follow a waterfall model and therefore they suffer the same criticisms of that approach. Finally, authors have suggested that any combination of approaches is untenable because each has different philosophies and therefore cannot be blended.

2 MULTIVIEW: A CONTINGENT FRAMEWORK

2.1 Background to Multiview

Multiview (Avison & Wood-Harper, 1990) was proposed as a framework for information systems development. Information systems development is perceived as a hybrid process involving computer specialists, who will build the system, and users, for whom the system is being built, with the help of a methodology. The methodology looks at both the human and technical aspects of information systems development. In this aspect and others, it has been greatly influenced by Soft Systems Methodology (Checkland, 1981) and ETHICS (Mumford, 1995) but has fused these ideas with those found in 'hard' methodologies, such as Yourdon Systems Modeling (Yourdon, 1993) and Information Engineering (Martin, 1989).

The approach adopted has been used on a number of projects, and the methodology itself has been refined using 'action research' methods (Checkland, 1981; Lewin, 1946; Susman & Evered, 1978; and Warmington, 1980), that is the application and testing of ideas developed in an academic environment into the 'real world'. It is a contingency approach in that it will be adapted according to the particular situation in the organisation. The authors are concerned to show that information systems development theories should be contingent rather than prescriptive, because the skills of different analysts and the situations in which they are constrained to work always has to be taken into account in any project. Avison and Wood-Harper (1986) describe Multiview as an *exploration* in information systems development. It therefore sets out to be flexible: a particular technique or aspect of the methodology will work in certain situations but is not advised for others.

The methodology includes many of the techniques used in other methodologies. The authors of Multiview claim, however, that it is not simply a hotchpotch of available techniques and tools, but an approach which has been tested and works in practice. It is also 'multi-view' in the sense that it takes account of the fact that as an information systems project develops, it takes on different perspectives or views: organisational, technical, human-orientated, social, economic and so on.

2.2 The original Multiview framework

The five stages of Multiview are as follows:

- Analysis of human activity
- Analysis of information
- Analysis and design of socio-technical aspects
- Design of the human-computer interface
- Design of technical aspects.

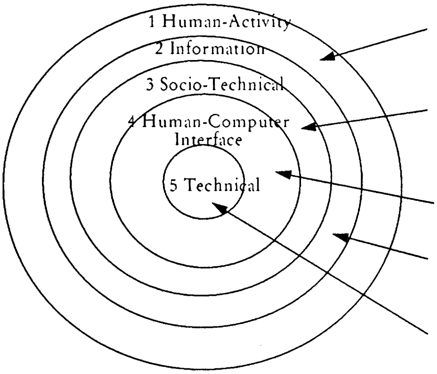
They incorporate five different views which are appropriate to the progressive development of an analysis and design project, covering all aspects required to answer the vital questions of users. These five views are necessary to form a system which is complete in both technical and human terms. The five stages move from the general to the specific, from the conceptual to hard fact and from issue to task. Outputs of each stage either become inputs to following stages or are major outputs of the methodology.

The authors argue that to be complete in human as well as in technical terms, the methodology must provide help in answering the following questions:

1. How is the computer system supposed to further the aims of the organisation installing it?
2. How can it be fitted into the working lives of the people in the organisation that are going to use it?
3. How can the individuals concerned best relate to the machine in terms of operating it and using the output from it?
4. What information system processing function is the system to perform?
5. What is the technical specification of a system that will come close enough to doing the things that have been written down in the answers to the other four questions?

Multiview attempts to address all these questions and to involve all the role players or stakeholders in answering these questions. The emphasis in information systems, it is argued, must move away from 'technical systems which have behavioural and social problems' to 'social systems which rely to an increasing extent on information technology'.

The distinction between issue and task is important because it is too easy to concentrate on tasks when computerising, and to overlook important issues which need to be resolved. Too often, issues are ignored in the rush to 'computerise'. Issue-related aspects, in particular those occurring at stage 1 of Multiview, are concerned with debate on the definition of system requirements in the broadest sense, that is 'what real world problems is the system to solve?'. On the other hand, task-related aspects, in particular stages 2-5, work towards forming the system that has been defined with appropriate emphasis on complete technical and human views. The system, once created, is not just a computer system, it is also composed of people performing jobs.



Q1- How is the information System supposed to further the aims of the organisation using it?

Q2 - How can it be fitted into the working lives of the people in the organisation using it?

Q3 - How can the individuals concerned best relate to the computer in terms of operating it and using the output from it?

Q4 - What information processing function is the system to perform?

Q5 - What is the technical specification of a system that will come close enough to meeting the identified requirements?

Figure 1 The Multiview framework (version 1)

One representation of the methodology is shown in figure 1. Working from the middle outwards we see a widening of focus and an increase in understanding the problem situation and its related technical and human characteristics and needs. Working from the outside in, we see an increasing concentration of focus, an increase in structure and the progressive development of an information system. This diagram also shows how the five questions outlined above have been incorporated into the five stages of Multiview.

The first stage looks at the organisation - its main purpose, problem themes, and the creation of a statement about what the information system will be and what it will do. It is based on SSM (mode 1), described in Checkland, 1981, using the techniques of rich picture building, CATWOE definition and the creation of root definitions, and conceptual models. Possible changes are debated and agendas drawn up for change. The second stage is to analyse the entities and functions of the problem situation described in stage one. This is carried out independently of how the system will be developed. The functional modelling and entity-relationship modelling found in most methodologies are suggested as modelling techniques.

The philosophy behind the third stage is that people have a basic right to control their own destinies and that if they are allowed to participate in the analysis and design of the systems that they will be using, then implementation, acceptance and operation of the system will be enhanced. Human considerations, such as job satisfaction, task definition, morale and so on are seen as just as important as technical considerations. This stage emphasises the choice between alternative systems, according to important social and technical considerations. The fourth stage is concerned with the technical requirements of the user interface. Choices between batch or on-line and menu, command or soft form interfaces are made. The design of specific conversations will depend on the background and experience of the people who are going to use the system, as well as their information needs.

Finally, the design of the technical subsystem concerns the specific technical requirements of the system to be designed, and therefore to such aspects as computers, databases, application software, control and maintenance. Although the methodology is concerned with the computer only in the latter stages, it is assumed that a computer system will form at least part of the information system. However, the authors do not argue that the final system will necessarily run on a large mainframe computer. This is just one solution, and many cases of Multiview in action show applications being implemented on a microcomputer.

2.3 The strengths and weaknesses of Multiview

Conventional information systems development methodologies have a number of weaknesses including:

- Narrow scope
- Rigidity in use
- Adherence to the waterfall model.

The authors argue that the first two aims are achieved in Multiview. The five parts of the approach encompass the aims of the organisation and how the information system can be fitted into the working lives of the people in the organisation that are going to use it, as well as addressing the user-computer interface, the functional requirements and the technical design. This is a much broader framework than that provided by more conventional methodologies.

A main tenet of Multiview is that it is a contingency approach, the techniques and tools suggested are to be used where appropriate and the phases and sub-phases may also be omitted or reduced in scope or executed in a different sequence than that shown in figure 1. Multiview is, however, not unstructured. An unstructured approach is offered by Benyon & Skidmore (1987) who suggest that information systems development should be a process of choosing techniques and tools as thought appropriate by the analysts at the time from a 'tool-kit'. Multiview provides a flexible framework and suggests (but does not put it stronger) a choice of techniques and tools at each phase in the development of a system. It allows the benefits of the experience and expertise embodied in good methodologies to be focused on the particular needs of the situation.

Although we have stated that phases might be omitted or reduced in scope or executed in a different sequence, the description of Multiview is in terms of 'layers in an onion' (as in figure 1) or as a series of five broad steps. However, this is described as an 'ideal type' which will guide the analyst who will redesign it for any practical situation. Nevertheless, the description gives the impression of a waterfall model, despite denials from the methodology authors using Multiview in practice. This led to difficulties where, for example, users required further explanation on how to go from stage 1 (essentially a description of the problem situation using SSM rich pictures, root definitions and conceptual models) to stage 2 (a combination of data modelling used in IE and process modelling used in STRADIS). A further refining of Multiview has led to another definition, and this is described in the next section. It is more explicitly an antithesis of the waterfall model.

2.4 The development of Multiview

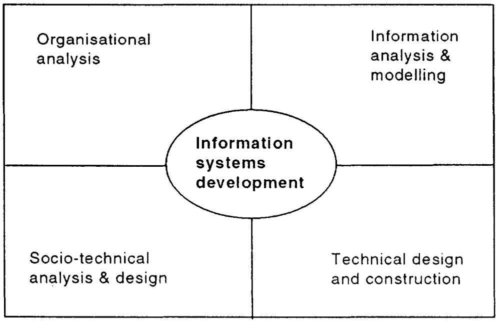

Organisational analysis
Information analysis & modelling
Information systems development
Socio-technical analysis & design
Technical design and construction

Figure 2 The Multiview framework (version 2)

In Multiview2 (Avison *et al.*, 1996), the five stages have been reduced to a four-box structure of organisational analysis, information analysis and modelling, socio-technical analysis and design, and technical design and construction. The proposed new framework for Multiview is given in figure 2 and it shows the four parts of the methodology mediated through the actual process of information systems development. The four parts of human activity systems analysis or organisational analysis (which examines organisational behaviours), socio-technical systems analysis and design (which examines work systems), and technical design and construction (which examines technical artefacts) are integrated through the information analysis and modelling stage which acts as a bridge between the other three, communicating and enacting the outcomes in terms of each other. In this way Multiview offers a systematic guide to any information systems development intervention, together with a reflexive, learning methodological process. Emphasis placed in each of the four parts of Multiview will change as the information system is being developed and contingent on the particular situation.

There are also differences in detail between the two versions of Multiview which reflect published research over the intervening years and more importantly experience in using Multiview during this period. Thus, for example, stakeholder analysis strengthens the conceptual analysis of SSM and ethical analysis in organisational analysis; there is a migration from structured methods to object-oriented analysis in information analysis and modelling; ethnographic approaches supplement ETHICS in socio-technical systems analysis and design; and prototyping, CASE, evolutionary and rapid development approaches are more strongly suggested in technical design and construction.

However, although the authors recommend a contingent approach to ISD, Multiview2 should not be used to justify random or uncontrolled development. The terms 'methodology' and 'method' tend to be used interchangeably, although they can be distinguished insofar as a method is a concrete procedure for getting something done while a methodology is a higher-level construct which provides a rationale for choosing between different methods (Oliga 1988). In this sense, an IS methodology, such as Multiview2, provides a basis for constructing a situation-specific method (figure 3), which arises from a genuine engagement of the analyst with the problem situation (Wastell 1996).

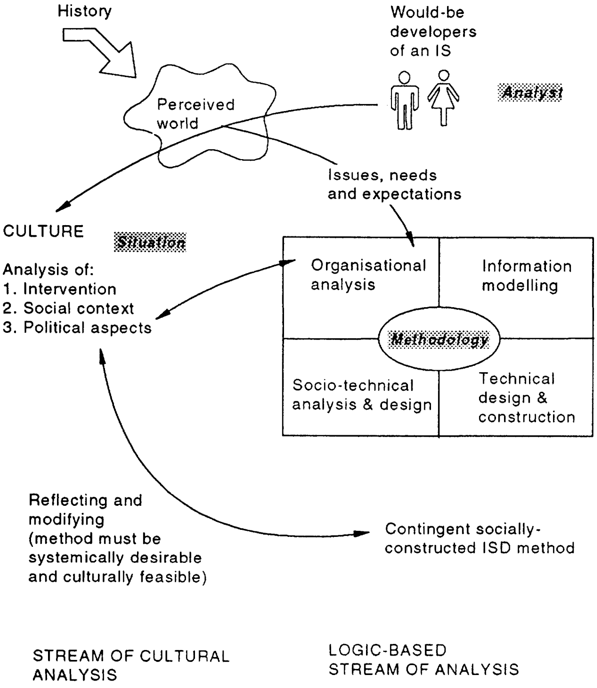

History
Would-be developers of an IS
Analyst
Perceived world
Issues, needs and expectations
CULTURE
Situation
Analysis of:
1. Intervention
2. Social context
3. Political aspects
Organisational analysis
Information modelling
Methodology
Socio-technical analysis & design
Technical design & construction
Reflecting and modifying (method must be systemically desirable and culturally feasible)
Contingent socially-constructed ISD method
STREAM OF CULTURAL ANALYSIS
LOGIC-BASED STREAM OF ANALYSIS

Figure 3 Constructing the information systems development methodology (adapted from Checkland & Scholes, 1990, Wood-Harper & Avison, 1992)

3 OBSERVATIONS AND CONCLUSIONS

3.1 Contingency

Strictly speaking, a distinction should be made in the criticisms of methodologies made earlier between poor application and use of a methodology on the one hand and an inadequate methodology itself. A defence made by a number of methodology vendors implies that the methodology is not being correctly implemented by some organisations. Whilst this may be true in some cases, it is not an argument that seems to hold much sway with methodology users. They argue that the two issues are much the same and for whatever reason they have experienced disappointments in the use of methodologies whether they represent improvements in the waterfall model (section 1.4) or new methodologies (section 1.5).

However, to respond to this by developing information systems without any methodology (section 1.3) is not the answer, as it will lead to the problems of poor control of the project, and management not getting value for money which were discussed in section 1.1 and, to some extent 1.2. Some authors (e.g. Benyon & Skidmore, 1987) advocate a 'tool-kit' approach, but it is argued elsewhere (Avison, Fitzgerald & Wood-Harper, 1988) that a tool-kit without any supporting advice and structures is indeed 'not enough' as it will lead to information systems that are likely to be idiosyncratic and difficult to maintain and therefore be of variable value.

One potential solution, outlined in section 1.6 and exemplified by Multiview which is described in section 2, is a more flexible approach, but within a framework, adaptable according to the characteristics of the project or domain. These contingent factors include the type of project, whether it is an operations-level system or a management information system, the size of the project, the importance of the project, the projected life of the project, the characteristics of the problem domain, the available skills and so on. As Mitroff & Linstone (1993) concluded, whilst any *inquiring system* can be used to generate evidence for any problem, it does not follow that each such system is equally valid or appropriate as a way of representing all kinds of problems.

A contingency approach to ISD is not new, indeed, was suggested in Davis (1982). Contingency is often seen as applying to techniques and tools only, however, it also applies to the general approach to information systems development, and that implies rejection of the waterfall model (except as a *special case* of applying a contingency approach). Multiview2 is more explicitly an antithesis of the waterfall model.

A contingent approach needs to be flexible enough to be appropriate for most situations. This implies that a broader, as well as more numerous, set of tools and techniques is available to the user of the contingency approach. But, it also implies that the approach carries within it the many 'philosophies' of the various approaches to information systems development suggested in section 1.4. Thus, such an overall approach might be a blend of human, technical, organisational, and other approaches to information systems development as found in the many methodologies proposed. Again, Multiview attempts to provide that broad-based framework. The method engineering movement also suggests a contingent and blended view of information systems development, but frequently a mainly (or even uniquely) technical view of the process. The authors of Multiview see information systems development as a social process containing technical aspects.

3.2 Information systems development as a social process

Defining an information system can be regarded as a social process with three aspects. These are the role of the systems analyst and the paradigm of assumptions constructed in practice; the political nature of the change process; and how methodologies are interpreted. These aspects are described in Wood-Harper & Avison (1992).

The theory about the role of the systems analyst and the paradigm of assumptions constructed in practice (Burrell & Morgan, 1979) can perhaps be explained best by giving examples of systems analysts in different situations. Four different stereotypical views of the systems analyst may be given as functionalist, interpretive, objective and subjective. The last three, to a greater or lesser degree, suggest that ISD is more of a social than a technical process. Roles, ideals and metaphors for each might be as follows:

- In the functionalist perspective, the information system consists of interactions which function independently of outside manipulation. The analyst assumes that the situation can be readily understood, indeed there is an assumption of rational behaviour by the actors which makes understanding easier. The systems are well controlled, can be well understood and can be formally defined. The systems analyst might be seen as technical expert, the ideals are objectivity, rigour and formality and a metaphor of the analyst might be a medical doctor. This is very much a technical and process view and one where ISD is seen as a technical rather than as a social process.
- In the interpretive perspective, it is assumed that the analyst is subjective and interprets the problem situation. The analyst hopes to understand the intentions of the actors in the situation. Participation and involvement will be the best way to obtain detailed information about the problem situation, and later to be able to predict and control it. The systems analyst might be seen as facilitator, the ideal might emphasise the importance of meaning and a metaphor of the analyst might be a liberal teacher.
- In the radical structuralist view, the situation will appear to have a formal existence but require radical change due to, for example, contradictory and conflicting elements. The systems analyst is assumed to be an agent for change and social progress, emancipating people from their socio-economic structures. The systems analyst might be seen as an agent for social progress, the ideals lean towards change of the socio-economic class structures and a metaphor of the analyst might be a warrior.
- Finally, in the radical humanist view, the situation is seen as external and complex. There is an emphasis on participation to enable a rapport between the actors and leads to emancipation at all levels, including the socio-economic and psychological. The systems analyst might be seen as change analyst, the ideals lean towards change of the socio-economic structures and psychological barriers and a metaphor of the analyst might be an emancipator.

Kling & Scacchi (1982) identified four perspectives within which problem solvers may view the content of the problem situation in which information technology is embedded. The importance of these perspectives for the information systems definition is that different strategies should be adopted based on the perspective embraced. The first is the formal rational perspective, which emphasises the formal organisational structure and procedures. With this perspective, we see the extreme of reductionist thought. Again, this is a traditional technical perspective. The second perspective, the structural perspective, includes considerations of the situation's formal subunits and recognises that communication must occur between them. The third perspective is the interactionist viewpoint which recognises that the pieces of the information resource are not independent nor formally defined. The social groups of interest cross intra-organisational and inter-organisational boundaries and are possibly in a constant state of flux. The process of change is founded on negotiation. The fourth perspective, organisational politics, assumes that interactions in the organisation are based on the political machinations and resulting manifestations of power. Again, as we progress through the four perspectives, we see less emphasis on the technical and structural and more emphasis on the social and potentially emancipatory.

Defining an information system can be thought of as metaphorical activity with, for example, the Multiview methodology as a non-prescriptive description of a real-world process. The essence of a metaphor is in the understanding and experiencing of one kind of thing in terms of another and, in this context, the methodology is a useful, epistemological device for the process of defining an information system (Lakoff & Johnson, 1980). This means that there is support from fieldwork that the Multiview methodology is a metaphor which is interpreted and developed in the situation. Consequently, the Multiview methodology can be thought of as being an 'open theory' where people close the theory in action.

3.3 The inter-disciplinary nature of information systems

Avison & Nandhakumar (1995) argue that information systems is a pluralist field and evidence this view through information systems development. There is a wide variety of approaches to information systems development and a large number of methodologies based on each of the general approaches. Longworth (1985) identifies over 300 information systems development methodologies. Wood-Harper & Fitzgerald (1982) discuss two basic differences of approaches as lying either within a systems paradigm or scientific paradigm, illustrated by soft systems method (Checkland, 1981) and structured analysis and design (DeMarco, 1979 and Gane & Sarson, 1979) respectively.

Avison & Fitzgerald (1995) widen the basis for comparison and suggest that information systems development methodologies can be compared on the basis of philosophy, model, techniques, tools, scope, outputs, practice and product, and they classify approaches within a number of broad themes including:

- Systems
- Strategic
- Participative
- Prototyping
- Structured
- Data
- Object-oriented.

None of these approaches can be described as different flavours to well accepted approach. They represent radically different approaches to information systems development and ways to perceive the information systems development process. They require different expertise: some emphasise people and stress the need for inter-personal skills; others require engineering skills and stress skills in the use of techniques; and yet others stress organisational issues. They represent different 'philosophies'.

If we consider the themes identified above as approaches to information systems development, disciplines relevant would seem to include, for example, computer science (prototyping tools and software engineering), mathematics (formal methods), sociology (participation) and business and management (planning). We may add applied psychology, economics, linguistics, politics, semiology, ethics, ergonomics, culture studies and probably others to the list of foundation disciplines. Information systems development has a multi-disciplinary nature, and technology and computing are by no means dominant. Unfortunately, it is clear that the majority of research into information systems development concentrates on the technical aspects, and this includes the languages and formalisms of the method engineering movement.

Mitroff & Linstone's 'Fifth Way of Knowing', which they call Unbounded Systems Thinking, subsumes all the other inquiring systems within it since every information system or inquiry presupposes all the others. In this way, each information system is mutually dependent upon the other and hence there is no sense in which they can be seen as having a distinct and separate existence, one from another. Unbounded systems thinking argues that complex problem solving requires the application of as many disciplines, professions, and branches of knowledge as possible, with each one employing different paradigms of thought.

ACKNOWLEDGEMENTS

I am grateful to my friends and colleagues who have contributed greatly to discussions which have led to this paper, in particular, Guy Fitzgerald, Richard Vidgen, Bob Wood and Trevor Wood-Harper.

REFERENCES

Avison, D. E. & Fitzgerald, G. (1995) *Information Systems Development: Methodologies, Techniques and Tools*. 2nd edition, McGraw-Hill, Maidenhead.

Avison, D. E., Fitzgerald, G. & Wood-Harper, A. T. (1988) Information systems development: a tool-kit is not enough. *Computer Journal*, 31, 4.

Avison, D. E. & Nandhakumar, J. (1995) The discipline of information systems: let many flowers bloom!, in *Information Systems Concepts: Towards a Consolidation of views* (eds. E. D. Falkenberg, W. Hesse, & O. Olivé) Chapman and Hall.

Avison, D. E. & Wood-Harper, A. T. (1986) Multiview - an exploration in information systems development. *Australian Computer Journal*, 18, 4.

Avison, D. E. & Wood-Harper, A. T. (1990) *Multiview: An Exploration in Information Systems Development*. McGraw-Hill, Maidenhead.

Avison, D. E., Wood-Harper, A. T., Vidgen, R. & Wood, R. (1996) *Multiview: A Further Exploration in Information Systems Development*, McGraw-Hill, Maidenhead.

Benyon, D. & Skidmore, S. (1987) Towards a tool-kit for the systems analyst. *Computer Journal*. 30, 1.

Bodker, S., Ehn., Kammersgaard, J., Kyng. M. & Sundblad, Y. (1987) A UTOPIAN experience: on design of powerful computer-based tools for skilled graphic workers, in: Bjerknes, G., Ehn. & Kyng. M. (eds.), *Computers and Democracy: A Scandinavian Challenge*, Avebury, Aldershot.

Booch, G. (1991) *Object Oriented Design with Applications*. Benjamin/Cummings, Redwood City, California.

Bullen, C. V. & Rockart, J. F. (1984) *A Primer on Critical Success Factors*. CISR Working Paper 69, Sloan Management School, MIT, Boston, Mass.

Burrell, G. & Morgan, G. (1979) Sociological Paradigms and Organisational Analysis. Heinemann, London.

Checkland P. B. (1981) *Systems Thinking, Systems Practice*. Wiley, Chichester.

Checkland, P. & Scholes. J. (1990) *Soft Systems Methodology in Action.* Wiley, Chichester.

Coad, P. & Yourdon, E. (1991) *Object Oriented Analysis.* 2nd ed., Prentice Hall, Englewood Cliffs, New Jersey.

Daniels, A. & Yeates, D. A. (1971) *Basic Training in Systems Analysis.* 2nd ed., Pitman, London.

Davis, G. B. (1982) Strategies for information requirements determination, *IBM Systems Journal*, 21, 2.

De Grace, P. & Stahl, L. (1993) *The Olduvai Imperative: CASE and the State of Software Engineering Practice.* Prentice Hall, Englewood Cliffs, New Jersey.

Hammer, M. & Champy, J. (1993) *Reengineering the Corporation: A Manifesto for Business Revolution.* Harper Business, New York.

Eva, M. (1994) *SSADM Version 4: A User's Guide.* 2nd ed., McGraw-Hill, Maidenhead.

Kling, R. K. & Scacchi, W. (1982) The web of computing: computing technology as social organization, *Advances in Computers*, 21.

Lakoff, G. & Johnson, M. (1980) *Metaphors We Live By*, The University of Chicago Press, Chicago.

Martin, J. (1991) *Rapid Application Development.* Prentice Hall, Englewood Cliffs, New Jersey.

Mitroff, I., & Linstone, H. (1993). *The Unbounded Mind, breaking the chains of traditional business thinking.* Oxford University Press, New York.

Mumford, E. (1995) *Effective Requirements Analysis and Systems Design: The ETHICS Method.* Macmillan, Basingstoke.

Quang, P. T. & Chartier-Kastler, C. (1991) *Merise in Practice.* Macmillan, Basingstoke (translated by D. E. and M. A. Avison from the French *Merise Appliquée.* Eyrolles, Paris, 1989).

Wastell, D. (1996) The Fetish of Technique: methodology as a social defence. *Information Systems Journal*, 6, 1.

Wood-Harper, A. T. & Avison, D. E. (1992) Reflections from the experience of using Multiview: through the lens of soft systems methodology, *Systemist*, 14, 3.