A COMPARATIVE SURVEY OF ACTIVITY-BASED METHODS FOR INFORMATION SYSTEMS DEVELOPMENT

Amanda Quek, Hanifa Shah
Faculty of Computing, Engineering and Technology, Staffordshire University, Beaconside, Stafford, United Kingdom.
Email: a.quek@staffs.ac.uk, h.shah@staffs.ac.uk

Keywords: Activity, Activity theory, Methods, Information systems, Survey

Abstract: The role of human factors and the importance of sociocultural and contextual issues in information systems (IS) development has long been recognised. However, these 'soft' details remain elusive and difficult to capture. Activity theory (AT) provides a framework with which to analyse and understand human behaviour in context. AT-based methods for IS development may therefore be a way forward. This paper presents a comparative survey of five AT-based methods. Each method is described, and its strengths and weaknesses briefly identified. The methods are then compared along nine key dimensions. As part of the findings, it is determined that most of the methods are selective in their use of AT, and are not sufficiently validated. Several correlations have also been noted across dimensions. Observations are presented on the limitations of existing methods, and suggestions are then made on possible ways forward.

1 INTRODUCTION & CONTEXT

This paper describes five activity theory (AT)-based methods for information systems (IS) development and makes a detailed comparison between them. The purpose of the comparison is to review the state of the art in AT-based methods, and to identify potential areas for improvement.

In the initial sections of this paper, a brief context of the problem is given, and an overview of AT is presented. Following that, five methods based on AT are described, identifying the general strengths and weaknesses of each. A detailed comparison along nine dimensions is made, and the results of the comparison are presented. Finally, suggestions for improvement are offered, and conclusions are drawn on the state of AT-based methods for IS development.

The importance of considering contextual and social issues in IS development has been well documented, for example Mumford and Ward (1968), Bignell and Fortune (1984), and Constantine (2001). However, the difficulties of capturing contextual information (Brown and Duguid, 1994), and the paucity of practical methods available for contextual analysis remains an ongoing concern in IS research. This has resulted in researchers turning to other fields of study in efforts to discover knowledge, appropriate skills, or develop methods that afford the capture of contextual and social factors that can contribute to IS development.

One example of an alternative approach to contextual analysis is AT. AT has been recommended for use in IS development as a guiding framework to elicit, analyse, understand and incorporate contextual features (Bødker, 1991, Kaptelinin, 1992, Kuutti, 1996, Hasan, 2001), and has been applied to various types of computer-based systems development.

Several AT-based methods have emerged for use in IS development. These methods are new, still evolving, and have emerged from diverse origins; hence as a way forward, this paper makes a detailed comparison between them and offers suggestions for possible areas of improvement.

We now look at a definition of 'method', and the reasoning behind the choice to use 'method' in this paper. Avison and Fitzgerald (1995) define method as a "recommended series of steps and procedures to be followed in the course of developing an IS". The major difference between a method and a methodology is that a methodology has an explicit underlying philosophy, whereas a method does not discuss philosophical aspects *(ibid)*. Therefore, we choose to use the term 'method' throughout this paper, as the underlying philosophy is not deemed an informative point of comparison. This is because the 'soft', human-focused nature of AT predisposes all AT-based work to have a similar philosophy in mind.

The following section is a brief introduction to AT and its body of principles and concepts.

2 ACTIVITY THEORY (AT)

Activity theory (AT) is a complex, abstract, theoretical framework that provides concepts and a vocabulary with which to analyse and understand human activity in context. The concept of 'activity' is an entire field of study (Wertsch, 1979), which cannot be detailed here. However, it can be said that a major tenet of AT is that human activity can only be understood within its sociocultural and historical context.

AT stems from the work of Russian psychologists Vygotsky and Leont'ev, in particular Vygotsky's (1934, 1978) theory of mediated activity, which argues that all purposeful human activity is accomplished through the use of physical and/or psychological tools. After Vygotsky, Leont'ev (1977, 1978) pioneered the concept of the hierarchical levels in activity, and explained that activity is always collective, never individual. AT was developed further by Engeström (1987), who produced the diagram of the activity system (Fig 1).

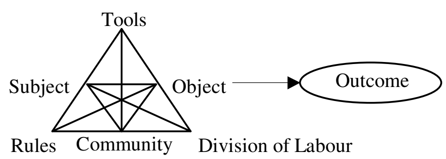

Figure 1: The Activity System (Engeström, 1987)

The activity system diagram consists of six interlinked components – the subject, the tools, the object (see section 2.2 for an explanation of the term), the community, the rules, and the division of labour. The subject refers to the actor or group of actors that is carrying out the activity. The subject uses tools to achieve an object, thereby turning the object into an outcome (fig. 1). A subject can carry out many activities, and each of the activities will have a different object. Activities can be distinguished from one another according to their different objects.

The tools refer to both the physical and psychological tools that are used by the subject in order to achieve the object. The rules refer to both the written and unwritten regulations which govern and constrain the subject's behaviour. The community encompasses all the actors who work with the subject and share the same object, and the division of labour refers to the way work is divided up between the subject and the community members.

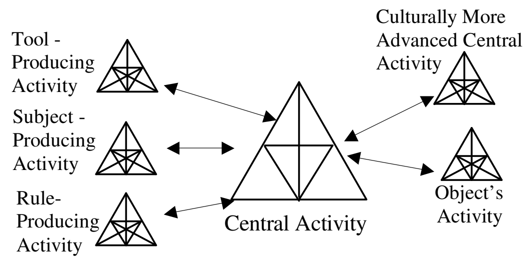

Figure 2: The Activity Network (Engeström, 1987)

The concept of 'contradiction' is an important one in AT. Engeström (1987) explains 'contradiction' as a conflict or clash within and between the components of the activity system, as well as conflict within and between activity systems. By identifying contradictions in an activity, we can identify areas where improvements can be made to work processes, and build these into the IS design.

Engeström (1987) also produced the activity network diagram (Fig 2), which illustrates the wider network of activities. In addition to the activity system and activity network diagrams, there are six guiding principles behind AT (Kaptelinin, 1992, Wertsch, 1979). Each principle is described briefly in the following sections, together with the way in which the principle can contribute to IS development.

2.1 Unity of consciousness and activity

This principle stresses that humans learn by doing, and that the human consciousness is formed by interaction with the external world (Kaptelinin, 1992). External activity is constantly informing and building the mental world, and vice versa – there is no separating the two. The implications for IS development are that in order to understand or influence human thinking and behaviour, we must look at the interaction between the person and their activity in context.

2.2 Object-oriented activity

The term 'object' as used in AT refers to both the 'objective' of the activity as a mental construct, and the physical or conceptual 'thing' that is worked on. Both of these evolve as the activity is carried out. Activity is described as being 'object-oriented' (not to be confused with the software engineering term), which means that every activity has a purpose (or object), and is carried out in order to achieve some outcome. Thus, it is essential to determine what the object of each activity is in order to design systems that support users in achieving what they need to do.

2.3 Mediation

The principle of mediation states that in every purposeful activity there will be tools involved, both physical (e.g. hammer and nails) and psychological (e.g. language and mathematics). A computer is a hybrid of the two: it is a physical tool that is used for psychological processing. AT views an IS as a tool for people who are carrying out some higher activity. This viewpoint shifts the emphasis from being technically oriented to user oriented. In order to design systems that support the work of humans, we need to understand what tools are currently used, how they have evolved, and how their functions could be supported by the system.

2.4 Hierarchical structure

This principle states that an activity can be decomposed into actions and operations. Each activity can be decomposed into actions, which are conscious steps that are taken to carry out the activity. Actions in turn can be decomposed into operations, which are subconscious. For example, from the point of view of an IS developer, the activity of 'developing an IS' can be decomposed into actions such as 'attending meetings', 'talking with users', and 'programming'. The action of 'programming' can be decomposed further into subconscious operations, such as using the keyboard and the mouse.

These levels are constantly in a state of flux. Conscious actions, if undertaken often enough, can become subconscious. Similarly, a subconscious operation can revert to a conscious action if a problem is encountered. An IS needs to support users' work at all three levels concurrently, to ensure a smooth transition through the levels.

2.5 Internalisation/externalisation

Internalisation is the process by which mental representations are formed by carrying out external actions. Externalisation is the opposite: where mental representations are manifested in external actions. Internalisation and externalisation are closely related to the way learning takes place. The implication of this principle is the need for the IS itself to support users in learning to use the system, to support the internalisation of system functionality, and to support users who are at different levels of expertise.

2.6 Development

This principle explains that activity can only be understood through analysis of its developmental transformations. All human activity is a result of historical development, and is constantly changing and re-forming. We need to analyse and understand how work has developed over time in order to fully grasp how work is done today. At the same time, the concept of development tells us that change and development are certain to occur whenever humans are involved. Therefore, as far as possible, we should ensure that the systems we design are equipped to handle change and development in work practice.

Within the field of IS development, AT has been widely used in computer supported collaborative work (Kuutti and Arvonen, 1992), computer supported collaborative learning (Gifford and Enyedy, 1999), and human-computer interaction (Bødker, 1991), and has been a major contributor to the Scandinavian tradition of systems design.

The following section describes AT-based methods in detail.

3 AT-BASED METHODS FOR IS

Following a review of the literature, five AT-based methods emerged that aim to operationalise the theoretical concepts of AT to produce practical methods for IS development. These methods are the ActAD method, the Activity Checklist, the AODM method, the Jonassen & Rohrer-Murphy framework, and the Martins & Daltrini framework. A comparison of these methods will be made from different dimensions, and conclusions are then drawn.

3.1 The ActAD method

The Activity Analysis and Development (ActAD) method was first developed by Korpela (1997). It is recommended for several purposes, namely for IS users (to improve their own work processes), IS developers (to analyse the IS users' work in order to design improved IS facilities for them), and IS researchers (to design improved IS methods and techniques for the developers). Korpela *et al.* (2000) focus in particular on the use of ActAD as a method for IS development.

ActAD provides a framework for IS developers to examine sociocultural features that can inform development of an IS. The first step guides developers to analyse the components of the activity that is to be supported by the system, and provides checklists of guiding questions with which to elicit these components. The surrounding activities are analysed in the second step. The activity system and the activity network diagrams are derived from those of Engeström (1987), and have been modified to become more graphical.

Step 3 in the method focuses on analysing the development of the central activity, which is broken down into 'History', 'Problems, and 'Potential'. A brief checklist of questions is provided with which to elicit each factor.

In the fourth step, the required new tools are to be developed, and processes are to be improved, based on the information elicited previously. The final step in ActAD involves disseminating the results, evaluating the process, and starting again, looping back to the problem analysis in Step 3.

A limitation of ActAD is the lack of a notation with which to document the third, fourth and fifth steps in the method. The sparseness of guidelines provided in these steps also calls for future development.

3.2 The Activity Checklist

The Activity Checklist was developed by Kaptelinin *et al.* (1999), and aims to enable researchers and designers to identify the contextual factors that can influence the use of computer technology in a real life setting, and to spot potential trouble areas that designers can address.

The checklist has two foci – design and evaluation, and subsequently there are two slightly different versions. Both versions of the checklist consist of four columns, based on four of the principles of AT. The column headings are: Means/ends (which relates to the hierarchical structure), Environment (related to object orientedness), Learning / cognition / articulation (related to internalisation / externalisation), and Development (named after the corresponding AT principle). The principle of mediation is said to permeate all four columns. Within each column, between 5 and 13 items are listed to guide the analysis. The developer is advised to generate their own questions based on the items listed, and a table of sample questions is provided.

The main weakness perceived with the checklist is the repeated usage of activity theoretical jargon. The checklist is also purely textual, and described at a high level of abstraction.

3.3 The AODM Method

The Activity-Oriented Design Method (AODM) was first detailed in Mwanza (2001) and is based on the models of Engeström (1987). AODM was developed further in Mwanza (2002), and is intended to contribute to the early phases of systems development, with an aim to support requirements capture, analysis, and design, focusing on human-computer interaction. The method consists of 6 stages, and 4 tools.

Stage 1 analyses the situation involved. For this stage, the first tool is provided - the ‘eight-step-model’. This is a list of 8 questions that guide the analysis of the activity and its components. Stage 2 involves modelling the situation, using the information obtained in Stage 1 with the activity system model (Engeström, 1987). Following that, stage 3 decomposes the activity to reduce complexity. The ‘activity notation’ tool is provided to assist in this stage. This tool details 6 ‘sub-triangles’ that can be analysed in order to decompose the activity.

Stage 4 is aided by a third tool, consisting of 6 general questions, which can be used to generate a wide range of research questions to analyse the interaction and relationships within and between the components of each subtriangle. This tool also elicits the presence of conflict within and between the components. In Stage 5, the research questions generated are used in data gathering, e.g. in interviews, questionnaires, or observation. Finally, Stage 6 involves interpreting and communicating the findings. For this a fourth tool is provided - the diagram for mapping operational processes. This tool presents the results of Stage 4 in illustration form, with clear visual indications of the research questions generated, as well as the areas of conflict that have become apparent, facilitating understanding of the process as well as the results.

A unique contribution of AODM is its development of subtriangles as further units of analysis. However, the application of AT in AODM appears to be in differing degrees, for example, the principle of mediation seems to be given more prominence than development.

3.4 The Jonassen & Rohrer-Murphy Framework

The framework of Jonassen & Rohrer-Murphy (1999) is a method for the design of constructive learning environments (CLE). There are six steps in the framework, each divided into a number of substeps. Each substep provides sample questions to be asked and actions to be taken.

The first step is primarily based on the principle of object-orientedness, and guides the developer in examining the subject’s (learner’s) goals to determine the purpose of the activity that the CLE is to support. In the second step, each component of the activity system is examined. The third step is chiefly based on the principle of hierarchical structure, and leads to the decomposition of the learner’s activities into actions and operations. The fourth step involves elicitation of the tools and other mediatory means that have been and could be used in the CLE.

Step 5 of the framework analyses the context, and the substeps within guide the analysis of the community, rules, and division of labour present in the activity. Finally, step 6 analyses the interaction and rules for the relationships that exist within and between the components of the activity system.

Each step of the framework is clearly based on individual principles and components from AT, but some principles influence all the sections. For example, the principle of development permeates the entire framework. The framework provides a large set of questions to be answered that cover diverse combinations of AT principles and components.

However, the level of granularity to which each question refers is not always clear. There is also repetition across the steps of the framework, as some questions are closely related.

3.5 The Martins & Daltrini Framework

The framework of Martins & Daltrini (1999) is an approach for requirements elicitation. There are three steps in the framework. The first step is to identify the activities to be supported by the target system. The second step is to identify the components of the activity system belonging to each activity, based on Engeström's activity system model (fig 1). The third step is to decompose each activity into actions and operations, based on the principle of the hierarchical structure of activity. This is done with the aid of a table with three columns: activity, action and operation. The list of actions and operations obtained can be used to derive requirements for the target system.

This framework focuses mainly on the principle of the hierarchical structure of activity. It provides little support for identifying activities, and there is little guidance provided for identifying the activity system components.

3.6 Summary of general strengths and weaknesses

Table 1 shows a summary of the general strengths and weaknesses observed in each of the methods.

Strengths	Weaknesses
ActAD	
Activity system model made more graphically oriented	Lack of a notation and sparse guidelines, in steps 3, 4, and 5
Activity Checklist	
Strong application of AT theoretical principles	Purely textual, abstract, high usage of AT jargon
AODM	
Unique contribution in using subtriangles as further units of analysis	Some principles given prominence over others e.g. mediation
Jonassen and Rohrer-Murphy	
Full application of AT principles, extensive list of analysis questions	Lengthy and sometimes repetitive
Martins and Daltrini	
Provides a notation for the decomposition of actions and operations	Does not provide guidance for identifying activities or components

Table 1: Strengths and Weaknesses of AT-based Methods

The strengths of each method indicate the contributions that are made, which future work could build upon. The weaknesses could indicate areas that need improvement. These have been discussed in detail in the previous sections.

4 COMPARISON OF THE METHODS

This section details the results of the comparison of the methods. There are many dimensions of comparison that could be taken, but due to space constraints, this paper looks at 9 key dimensions. We propose that a comparison of AT based methods requires several different dimensions to those used in studies that compare IS methods. Traditional factors for comparison between IS methods are such as life cycle coverage, underlying philosophy ('systems' or 'science'), 'structuredness', user role, and techniques used (data flow diagrams, dialogue design, entity modelling, etc) e.g. Tudor and Tudor (1997). Some of these dimensions (e.g. the development phases supported) are suitable for our purpose. However, the main focus in comparing AT-based methods is on the way that AT is applied and made practical by the method. To reflect this, in addition to traditional dimensions for comparison, we have chosen to compare the extent to which the method uses AT visual models, the AT principles that informed the method, the elicitation of 'contradiction' and historical background, and the provision of analysis questions. Because these methods are new, we also compare the way the method was developed and validated.

- Development phases supported
- Area or type of system supported
- AT visual models used
- AT principles that informed the method
- Elicitation of 'contradictions' in the activity
- Elicitation of the historical background
- How the method was developed
- Provision of analysis questions
- Validation of the method

There are other factors that could be used for comparison, but due to space constraints, only 9 key dimensions are described here.

4.1 Development phases supported

Table 2 shows that each method supports different phases of development. No method covers all phases of development. The phases that are strongly supported by methods are domain analysis, requirements elicitation and design. There is less support for the phases of evaluation and interface design. However, this does not mean that AT is less suitable for use in these phases, or in phases that are as yet unsupported by methods (such as the implementation phase). Rather, this testifies to the need for further research into producing and testing

AT-based methods that can be used throughout the IS development process.

	Domain Analysis	Requirements Elicitation	Design	Interface Design	Evaluation
ActAD	←→				←→
Activity Checklist			←→		←→
AODM	←——	——	——	——→	
Jonassen	←→	partial	←→		
Martins & Daltrini		←→			

Table 2: Development phases supported

4.2 Area or type of system supported

AT-based approaches have been proposed for use in different areas of computing, and for several types of systems design. Table 3 details the area of study and the type of system supported by each method. Two of the methods were designed for specific areas of application, namely, the AODM method, which is aimed specifically towards human computer interaction, and the Jonassen & Rohrer-Murphy framework, which is oriented towards designing computer supported collaborative learning. The other three methods are for general systems development.

	Area or type of system supported
ActAD	General
Activity Checklist	General
AODM	Human Computer Interaction
Jonassen	Computer Supported Collaborative Learning
Martins & Daltrini	General

Table 3: Area or type of system supported

4.3 AT visual models used

Two of the visual models provided by AT are the activity system and activity network (Engeström, 1987). Table 4 shows that three of the methods use the activity system model, and one of these three also uses the activity network.

	Activity System	Activity Network	Sub-triangles	Table of activities, actions, and operations
ActAD	Yes	Yes	No	No
Activity Checklist	No	No	No	No
AODM	Yes	No	Yes	No
Jonassen	No	No	No	No
Martins & Daltrini	Yes	No	No	Yes

Table 4: AT visual models used

Two methods provide their own visual models– the AODM method provides a graphical representation of the subtriangles within the activity system, and the Martins & Daltrini method provides a table to record the breakdown of activities, actions and operations. Two methods, namely the Activity Checklist and the Jonassen & Rohrer-Murphy framework, do not use any visual models.

4.4 AT principles applied

Table 5 shows the AT principles that are explicitly applied by each method. Upon comparison, the Activity Checklist and the Jonassen & Rohrer-Murphy methods produce the broadest application of the AT principles, that is 5 for the Checklist and 6 for the Jonassen & Rohrer-Murphy framework. The Martins and Daltrini framework only applies one principle- the 'hierarchical structure'. The only method that explicitly applies the principle of the 'unity of consciousness and activity' is the Jonassen and Rohrer-Murphy framework. The choice of which principles to apply seems to depend on the author of the method, and for the most part it is unclear how those choices were made.

	Unity of C. and A.	O-O	Med.	Hier. Str.	I / E	Dev.
ActAD		✓				✓
Activity Checklist		✓	✓	✓	✓	✓
AODM		✓	✓			
Jonassen	✓	✓	✓	✓	✓	✓
Martins & Daltrini				✓		

Table 5: AT principles applied

Key to AT Principles:

Unity of C. and A. – Unity of Consciousness and Activity
O-O – Object-Orientedness
Med. – Mediation
Hier. Str. – Hierarchical Structure
I / E – Internalisation/Externalisation
Dev. – Development

4.5 The elicitation of 'contradictions'

Identifying contradictions produces a significant contribution towards designing systems that improve the way work is done. Table 6 shows that more than half of the methods carry out some form of analysis of the contradictions existing within the work activity.

	Elicitation of 'contradictions'
ActAD	Yes
Activity Checklist	No
AODM	Yes
Jonassen	Yes
Martins & Daltrini	No

Table 6: Elicitation of 'contradictions'

4.6 The elicitation of the historical background of the activity

Understanding the historical background is important if we are to understand how the activity has developed over time. Table 7 shows that only one method includes the analysis of the historical aspect of activity: the ActAD method.

	Elicitation of historical background
ActAD	Yes
Activity Checklist	No
AODM	No
Jonassen	No
Martins & Daltrini	No

Table 7: Elicitation of historical background

4.7 Analysis questions provided

AT can be seen as a framework for analysis of activity. In order to carry out an AT analysis, information has to be elicited from the practitioners of the activity. Therefore, specific leads or questions for analysis are required.

	Analysis questions provided
ActAD	Partial
Activity Checklist	Partial/Yes
AODM	Yes
Jonassen	Yes
Martins & Daltrini	No

Table 8: Analysis questions provided

Table 8 shows that two of the methods provide clear sets of analysis questions; in fact, the Jonassen & Rohrer-Murphy framework consists entirely of sets of questions. The AODM method also provides questions, and has a dedicated step for generating research questions. The Activity Checklist provides sample questions, but encourages the analyst to create their own. The ActAD method provides questions for its initial three steps, but the questions for the third step are at a general level, and are of limited use. Finally, the Martins & Daltrini method does not provide any questions.

4.8 Development of the method

The way the method was developed can provide us with information on the orientation of the method – whether it is developed theoretically or empirically.

	Development of the method
ActAD	At desk
Activity Checklist	At desk
AODM	Empirical
Jonassen	At desk
Martins & Daltrini	At desk

Table 9: Development of the method

Table 9 shows that four of the methods were developed 'at desk', in other words, based solely on the theory. Only the AODM method was developed empirically, in a real world commercial environment.

4.9 Validation of the method

The importance of method validation cannot be underestimated, particularly for AT-based methods, which are new. Rigorous testing is required to ensure that AT based methods are able to provide quality of output.

	Validation of the method
ActAD	Two small experiments, neither real life
Activity Checklist	Self validation by the authors
AODM	Two industrial organisations over two years
Jonassen	None
Martins & Daltrini	One interview

Table 10: Validation of the method

Table 10 shows that only the AODM method was tested in a real life setting, in two industrial organisations. The ActAD method, the Checklist and the Martins & Daltrini method were validated in small experiments, and the Jonassen & Rohrer-Murphy framework does not provide any information that it has been validated.

4.10 Findings from the Comparison

The findings of the comparison include a number of correlations across the dimensions that were used. One finding is that a broad use of AT principles seems to correlate with less use of AT visual models, and vice versa (Sections 4.3 and 4.4). For example, both the Activity Checklist and the Jonassen & Rohrer-Murphy methods indicate high usage of the principles of AT, and non-usage of visual models, while the ActAD, AODM and Martins & Daltrini methods indicate low usage of the principles and the use of two out of four visual models.

Several methods are also selective in their application of AT principles, and it is often unclear how these choices were made. To avoid the arbitrary application of AT, it is important that justification be provided in future by methods that selectively apply AT concepts.

It is also found that methods developed 'at desk' were not subjected to as much validation as the empirically developed method (Sections 4.8 and 4.9). Further empirical research is necessary in order to validate the methods. More research is needed to produce methods that support wider coverage of development phases.

5 CONCLUSION

This paper has described five AT-based methods for IS development, and has made a comparison between them along nine dimensions. The general strengths and weaknesses of each method were also described, which indicate areas of strong contribution as well as potential for improvement for each method. As part of the findings, several correlations have emerged between the dimensions of comparison. It is found that within the AT-based methods that have emerged from the survey, there is a lack of comprehensive treatment, regarding coverage of development phases as well as coverage of AT concepts. It is also found that only one of the methods has been validated in a real life systems development.

In the light of the findings, we suggest that existing methods need to be thoroughly tested and documented through empirical studies in practice. Further work is also needed, both to improve existing methods, as well as to produce new methods for making AT concepts applicable in practical development scenarios.

ACKNOWLEDGEMENTS
This research is funded by the Tracker project at Staffordshire University (EPSRC Grant No. GR/R12183/01).

REFERENCES

Avison, D. E. and Fitzgerald, G. (1995) *Information Systems Development: Methodologies, Techniques and Tools,* McGraw Hill.

Bignell, V. and Fortune, J. (1984) *Understanding Systems Failures,* Manchester University Press.

Bødker, S. (1991) *Through the Interface - A Human Activity Approach to User Interface Design,* Lawrence Erlbaum Associates.

Brown, J. S. and Duguid, P. (1994) Borderline Issues: Social and Material Aspects of Design. *Human-Computer Interaction,* **9,** 3-36.

Constantine, L. L. (2001) *The Peopleware Papers: Notes on the Human Side of Software,* Prentice Hall.

Engeström, Y. (1987) *Learning by Expanding: An Activity Theoretical Approach to Developmental Research*.

Gifford, B. and Enyedy, N. (1999) Activity Centred Design: Towards a theoretical framework for CSCL. In *3rd International Conference on Computer Support for Collaborative Learning*.

Hasan, H. (2001) An Overview of Different Techniques for applying Activity Theory to Information Systems. In *Information Systems and Activity Theory: Theory and Practice*(Ed, Hasan, H.) University of Wollongong Press.

Jonassen, D. H. and Rohrer-Murphy, L. (1999) Activity Theory as a Framework for Designing Constructivist Learning Environments. *Educational Technology Research and Development,* **47,** 61-79.

Kaptelinin, V. (1992) Activity Theory: Implications for Human-Computer Interaction. In *Context and Consciousness: Activity Theory and Human-Computer Interaction*(Ed, Nardi, B.).

Kaptelinin, V., Nardi, B. A. and Macaulay, C. (1999) The Activity Checklist: A Tool For Representing the "Space" of Context. *Interactions*.

Korpela, M. (1997) Activity Analysis and Development in a Nutshell. http://www.uku.fi/atkk/actad/nutshell.html.

Korpela, M., Soriyan, H. A. and Olufokunbi, K. C. (2000) Activity Analysis as a Method for Information Systems Development. *Scandinavian Journal of Information Systems,* **12,** 191.

Kuutti, K. (1996) Activity Theory as a Potential Framework for Human-Computer Interaction Research. In *Context and Consciousness: Activity Theory and Human-Computer Interaction*(Ed, Nardi, B.) Massachusetts Institute of Technology.

Kuutti, K. and Arvonen, T. (1992) Identifying Potential CSCW Applications by Means of Activity Theory Concepts: A Case Example. In *CSCW '92*ACM.

Leont'ev, A. N. (1977) Activity and Consciousness. http://www.marxists.org/archive/leontev/works/1977/leon1977.htm.

Leont'ev, A. N. (1978) Activity, Consciousness, and Personality. http://www.marxists.org/archive/leontev/works/1978/intro.htm.

Martins, L. E. G. and Daltrini, B. M. (1999) An Approach to Software Requirements Elicitation Using Precepts from Activity Theory. In *14th IEEE International Conference on Automated Software Engineering*, pp. 15-23.

Mumford, E. and Ward, T. B. (1968) *Computers, Planning and People,* BT Batsford Ltd, London.

Tudor, D. J. and Tudor, I. J. (1997) *Systems Analysis and Design*
A Comparison of Structured Methods, Macmillan Press.

Vygotsky, L. S. (1934) *Thought and Language,* Massachusetts Institute of Technology, Cambridge, Massachusetts.

Vygotsky, L. S. (1978) *Mind In Society,* Harvard University Press, Cambridge, Massachusetts.

Wertsch, J. V. (1979) The Concept of Activity in Soviet Psychology: An Introduction. In *The Concept of Activity in Soviet Psychology*(Ed, Wertsch, J. V.) M E Sharpe, Inc.