Mapping Activity Theory Diagrams into i* Organizational Models

Genésio Gomes da Cruz Neto
Faculdade Integrada do Recife (FIR).
Av. Eng. Abdias de Carvalho, n.º 1678 - Madalena Recife - PE - CEP: 50720-635

Alex Sandro Gomes
Cento de Informática – Universidade Federal de Pernambuco (UFPE)
Caixa Postal 7851 – 50.732-970 – Recife – PE – Brasil

Jaelson Brelaz de Castro
Cento de Informática – Universidade Federal de Pernambuco (UFPE)
Caixa Postal 7851 – 50.732-970 – Recife – PE – Brasil

ABSTRACT

Modern requirement engineering approaches divide the elicitation process in two different stages: one focused on analyzing the context where the system-to-be will be used and another centered on designing software solutions appropriated to the context modeled. An adequate framework for assisting context analysis is offered by the Activity Theory, a philosophic and interdisciplinary structure to study different forms of human practice that adopts the activity as the basic unit of analysis. However, there are still no methods for integrating context analysis based on Activity Theory and traditional requirement specifications techniques. In a previous work, the authors presented a requirement engineering process that integrates ethnographic analysis based on Activity Theory with requirement specification techniques based on organizational modelling. In this work we present an evolution of the process proposed by including a set of mapping guidelines to systematically transform Activity Theory diagrams into i* based organizational models. Moreover, we apply the guidelines in the development of a virtual project based learning environment.

Keywords: Ethnography, Context Analysis, Organizational Modelling and Activity Theory

1. INTRODUCTION

The analysis of human practices, and its social relations, using techniques such as Ethnography [19], is mentioned in the literature as one of the important methods for capturing software requirements. Activity Theory is a theoretical-based context analysis technique that can anchor the ethnographer descriptive work, calling his attention to the individual and social elements of human activities [9].

For more than a decade, Activity Theory has been a recognized framework for enhancing design practices in Computer Supported Collaborative Work (CSCW) and related fields of Human-Computer Interaction (HCI) [1, 12, 13]. However, in most of the applications it has been deployed as an analytical framework, not oriented towards requirement engineering specifications [4, 7, 11, 12, 13, 18].

Present requirement engineering approaches, on the other hand, do not provide techniques for modeling human practices with neither the same level of detail, nor the same theoretical basis offered by the Activity Theory. Therefore, there is currently a need for techniques that could integrate ethnography analysis based on Activity Theory with present requirement engineering approaches.

Our problem is focused on a requirement engineering methodology which integrates Activity Theory analysis with i* [20] organizational modeling. In this work, we have enhanced the process proposed in [5] that uses ethnographic analysis based on Activity Theory to guide and complement the TROPOS [3] early and late requirement phases. In particular, we describe guidelines that can be used to derive i* organizational models from Activity Theory diagrams during the early requirement phase. The guidelines can be easily applied by a systematic analysis of the participants dependence relations occurred in and between activities.

This paper is structured as follows: section 2 discusses related works. The Activity Theory framework is described in section 3. In section 4, we present the requirement engineering methodology that integrates context analysis based on Activity Theory with the TROPOS methodology. The central part of the paper is detailed in section 5, which introduces the mapping guidelines used to systematically transform Activity Theory diagrams into i* organizational models. The guidelines are explained by presenting its application in the context analysis of a project based learning system. Conclusions and future works are described in section 6.

2. RELATED WORKS

Viller and Sommerville in [19] integrates ethnographic analysis with requirement engineering techniques. The authors introduce a method that integrates human practices analysis based on social viewpoints with the Use Case [15] requirement specification technique.

Martins´s thesis [10] is centred on a requirement elicitation methodology based on Activity Theory. His focus is on using activity diagrams based on Activity Theory to specify software requirements. He demonstrates that activity models can describe a scenario using a set of information richer than Use Case diagrams. The main differences between Martins approach and our proposal are described in sequence.

In our methodology we separate the elicitation process in context analysis (early requirement) and future system analysis (late requirement). Dividing the requirement phases makes it possible the use of Activity Theory for providing a theoretical framework for context analysis in early requirements, and for guiding the design of system alternatives in late requirements that are adapted to the modeled context. The adoption of these phases can avoid the existence of a frequent requirement engineer problem of modeling the system scope and functionality before having a clear understanding of the user needs. Martins´s methodology does not have a clear division between the referred requirement phases, which can bring difficulties to know if the activities described are related to context modeling or system specification.

We recognize that Activity Theory diagrams can be used to specify requirements, as described in Martins´s work. However, the flexibility richness of Activity Theory makes it useful for context analysis, but can bring difficulties to specify more formal aspects such as system requirements. Moreover, using activity diagrams to describe requirements can imply in additional learning efforts by software developers [17].

Therefore, our approach follows Korpella, Sorivan and Olufokunbi proposal [7] as in: "The more technical aspects of Information system development we approach, the less activity analysis and development can contribute, and the more we need methodologies adapted from software engineering. There is currently a gap between the less formal, innovative methods like activity analysis, and the strictly formal methods of Software Engineering".

3. ACTIVITY THEORY (AT)

Activity Theory [8] is a broad theoretical framework for describing the structure, development, and social context of human activities. It has roots in the historical-cultural soviet psychology founded by Lev Vygostsky, A. N. Leont´ev and A. N. Luria [12]. According to this theory, an activity is the way a subject (either an individual or a group) moves towards an object with the purpose of attaining certain results or certain objectives. Activity objects can be a concrete thing (such as a program) or something more abstract (for example, an idea).

Mediation tools, such as a text editor or an e-mail system, are artifacts used to support the object transformation into a result. Tools can be used to manipulate and understand the object, or to improve the communication and motivation of the activity participants.

Human practices are always included in a social context. In Activity Theory the systemic relationships between the subject and its environment are represented by the concepts of community, rules, and labor division. The community is formed by all subjects interested in activity development (usually called by Stakeholders in software engineering community). Rules are conventions for social relationships established by the community. Labor division refers to the form of community organization for the process of transformation of an object into a result. Figure 1 illustrates the systemic model proposed by Engeström [6] that shows the relationships between the structuring elements of the activity.

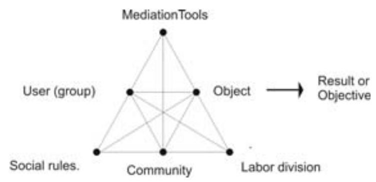

Figure 1. The Engeström systemic model.

Activities can be also defined in different levels: activity consists of actions or chains of actions. Participating in an activity is performing conscious actions that have an immediate and defined goal, or objective. Actions are linked to each other in one activity by the same overall motivation of transforming the activity object into an outcome. The activity motivation is the characteristic that differentiates and relates activities and actions.

Actions can be broken in lower level sub-actions, which, in turn, have sub-goals. Lower level actions carried automatically are operations. Actions turn into operations when they become routine and unconscious. Operations are well-defined habitual routines used as answers to conditions. The activity levels, and their dynamics, are showed graphically by Figure 2.

Human practice activities are not isolated. Real life situations always involve an intertwined and connected web of activities that are usually specified using an activity diagram. Figure 2 in section 5 presents an example.

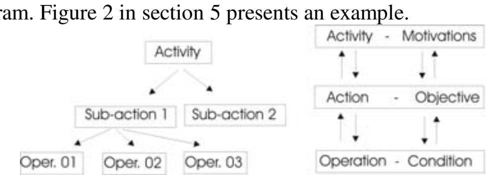

Figure 2. Activity levels.

4. INTREGATING AT AND I*

In this work, we present an evolution of the requirement engineering process proposed in [5] which is centred on an extension of the context analysis phase (Early Requirement) and the system analysis phase (Late Requirement) of the TROPOS [3] methodology. The paper presents a set of mapping guidelines that systematically transform Activity Theory diagrams into i*-based [20] organizational models.

TROPOS was chosen because it uses i*-based organizational models, not just for the requirement phases, but also as a foundation to the entire software development process. Besides, there are also guidelines to support the generation of UML [15] Use Cases and Class diagrams from i* organizational models [16, 2]. Therefore, we intend to show that context analysis based on Activity Theory can be used as a front-end for nowadays requirement specification approaches.

The process is started with an ethnographical context analysis of the human practices that result in transcribed interviews, recorded interactions and participant observation field-notes. In a second stage, there is a qualitative analysis where the collected information are then sorted and classified with Nud*ist software [14]. From the classified information, the activities (with its actions and operations) are described using the Engstrom model [6].

The context analysis phase (Early Requirement) ends with the generation of i* organizational models from the activities modeled. The transformation of activity diagrams into i* models are based on mapping guidelines that analyses the dependencies between subjects in and between activities. These guidelines are the central point of this article and are demonstrated in the next section.

In Late Requirement, the activity diagrams are used to guide the design of systems that can support the human practices modeled. We believe that systems are adapted to the modeled context if they can really contribute to activity development.

5. MAPPING AT DIAGRAMS INTO I* MODELS

To model the organizational environment (including their systems), the TROPOS methodology uses two types of i* models: the Strategic Dependency Model and the Strategic Rationale Model. In sequence, we describe the mapping guidelines used to generate these models by showing its application in the context analysis phase of a project based learning system.

First, is presented the Activity Diagram obtained as a result of an ethnographical analysis of the teaching-learning activity. This diagram is used as a starting point for the application of the mapping guidelines. The emphasis of the ethnographical observations was on a group of 4 students, together with the lecturer, in a software engineering course that employs the project based learning educational methodology. The analysis was based on in-loco observations and tape-recorded student interactions during 5 classes of one and a half hour each.

The Activity Diagram

The activity diagram of Figure 2 specifies four activities identified in the ethnographical analysis: *Construction of Project Proposals*, *Project Formation*, *Project Development* and *Project Evaluation*. In the first activity the lecturer works individually in the propositions of projects. The proposed projects are used in the *Project Formation* activity, where the students and the lecturer together define the projects to be developed. The *Project Development* activity in turn uses the defined projects as object. At end, in the *Project Evaluation* activity, the lecturer evaluates the projects. Each activity in figure 2 is described using the Engeström [6] systemic model. For example, observing the *Project Formation* activity, is possible to infer the activity is motivated by the transformation of *Project Proposals* into real projects using a Whiteboard, a Text-Editor and an E-mail system as mediation tools. The participants of this activity adopt a social rule that each project must have a manager. Besides that, the diagram illustrates the community interested in this activity, which includes the course coordinator. The activity description ends with the division of work where the lecturer is responsible for presenting the project proposals.

Next, each activity is detailed by specifying their actions and operations. The level of detail used depends on the particularities of each system. At least, it should provide sufficient information to show how each system requirement would support the activities modeled. Table 1 illustrates the actions present in the *Project Formation* and *Project Development* activities

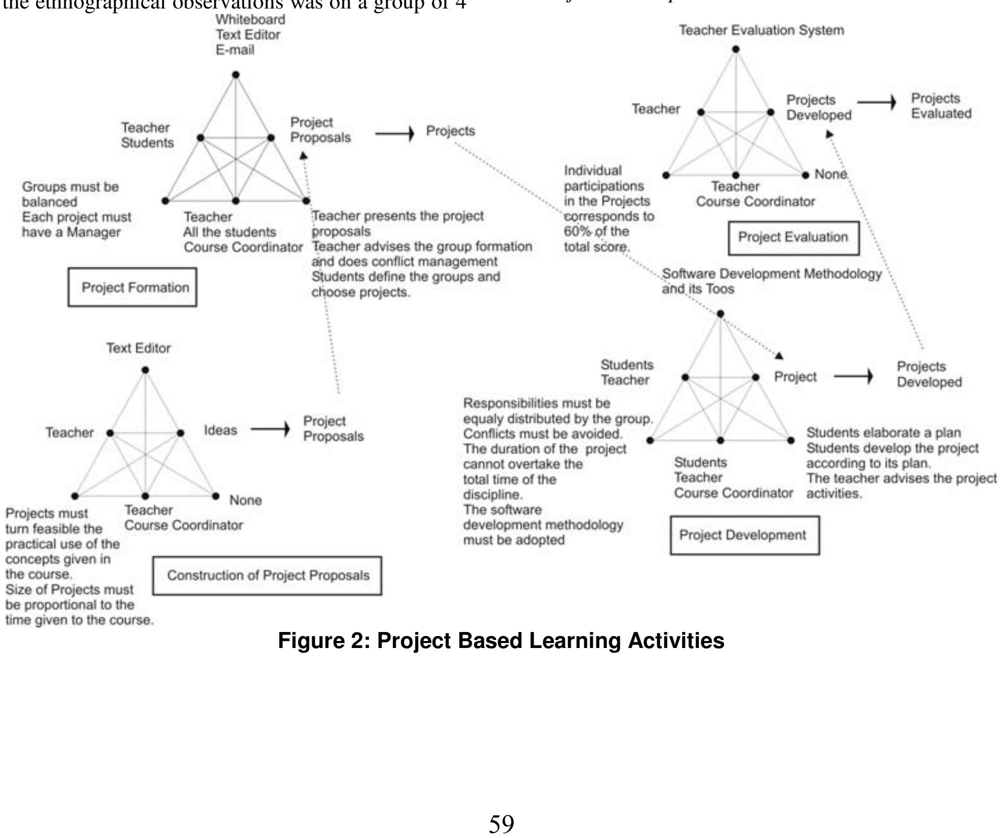

Figure 2: Project Based Learning Activities

Table 1: Decomposing Activities into Actions

Activity	Actions
Project Formation	Lecturer presents project proposals
	Students indicate preferences for specific proposals.
	Lecturer does conflict management
Project Development	Students define project plan.
	Students interview clients.
	Students document system requirement.
	Students develop system prototype
	Students present projects developed
	Lecturer motivates students
	Lecturer advises students

Depending on the system being designed, some actions can be decomposed into sub-actions and operations. Table 2 shows an example of how the action of "*Students define project plan*" can be divided in sub-actions.

Table 2: Decomposing Actions into Sub-Actions

Action	Sub-Actions
Students define project plan	Definition of project activities and phases
	Definition of project timetable
	Division of responsibilities in each project activity.

Generating the Strategic Dependence Model

The Strategic Dependency Model is a graph composed by the environment actors and their dependence relations. The dependence relations modeled can be of various types, including Goal Dependence, Soft-Goal Dependence (goal whose achievement evaluation is subjective), Task Dependence e Resource Dependence.

This section explains the guidelines that can be used to generate, in a systematic way, Strategic Dependence Models from Activity Theory Diagrams. The guidelines used to generate Strategic Rationale Models are presented in the next section.

Figure 3 shows the Strategic Dependence Model generated from the Activity Diagram of Figure 2, as well as the guidelines used to generate each element of the model. In sequence, each of these guidelines is explained in detail by showing how they can be used to generate the elements of this Strategic Dependence Model.

Guideline 1 - Actors: every subject is modeled as an actor in the organizational model, provided that one of the above conditions is satisfied:

- The subject depends on another person to work in a given activity.
- The subject depends on a resource provided by another activity.
- The goal of the subject is to produce something to be used by actors of other activities.

Observing the activities modeled in Figure 2, we identify as subjects: the students and the lecturer. Applying the guideline 1 we have the adoption of these subjects as actors in the model of Figure 3.

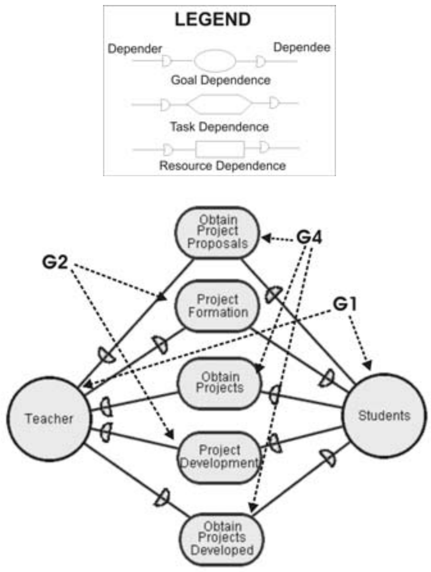

Figure 3: Strategic Dependence Model

Guideline 2 – Dependencies Between Subjects in a Single Activity: when one activity is carried out by two subjects, a dependence relation is generated between them whose name is the same given to the activity. The subject more responsible for the activity achievement is the Depender, while the one who has only a support role is classified as Dependee. The dependence type is obtained by Guideline 3, which analyses the characteristics of the task and the autonomy of the Dependee.

Guideline 3 – Dependence Type: if the *Dependee* has autonomy to achieve his tasks a *Goal* Dependence Relation is generated. In case of the Dependee does not have autonomy, the dependence type is of *Task* Dependence Relation. But if the *Dependee* is responsible for making an existing resource available, then *Resource* Dependence Relation is generated. Finally, if the *Dependee* has autonomy to obtain a goal whose achievement evaluation is subjective then the dependence relation is of type *Soft-Goal*. This classification is according to [20] which defines the inherent characteristics of the *Goal*, *Soft-Goal*, *Task* and *Resource* dependence relations.

Applying Guidelines 2 and 3 to the activity diagram of Figure 2, we have the generation of the Project Formation and Project Development goal dependence relations in Figure 3. Observing the labour division of Project Formation, we can realize that the lecturer is the main responsible for this activity. Therefore, we have the generation of goal dependence relation from the lecturer to the students (Teacher → Project Formation → Students). The Project Development activity, on the other hand, has the students as the main responsible for its success. In this situation, the students are the Dependers of the lecturer (the Dependeee).

Guideline 4 – Dependencies Between Subjects Across Activities: when a subject X of activity A uses something (object, social rule or mediation tool) provided by another subject Y of activity B, then a dependence relation is drawn in the organizational model between the subject X and Y. The subject X (the *Depender*) depends on something produced by subject Y (*Dependee*) to achieve the success in his activity. The dependence relation created receives the name given to the element being transferred preceded by the word "Obtain …". The dependence type is deduced using Guideline 3.

The goal dependence relations of Obtain Project Proposals, Obtain Projects and Obtain Projects Developed present in Figure 3 were created according to Guideline 4. In the activity diagram it is possible to establish a dependence relation between the students of the Project Formation activity and the lecturer, subject of the Construction of Project Proposal (the students depend on the Projects proposals produced by lecturer). In Project Development, the students depend on the lecturer to obtain the projects, since the lecturer participates in the Project Formation activity. On the other hand, the lectures depend on the students of the Project Development activity to evaluate the projects in the Project Evaluation activity.

Generating The Strategic Rationale Model

The Strategic Rationale Model allows one to model the reasons for the dependence relations presented in the Strategic Dependence Model. In another words, this model describes how the subjects internally act to satisfy their external dependence relations. This section explains the guidelines used to derive the Strategic Rationale Model.

Figure 4 shows the *Strategic Rationale Model* generated by the application of these guidelines on the activity diagram of Figure 2. In sequence, these guidelines are presented by demonstrating how they can be used to generate the elements of Figure 4.

According to Activity Theory, each subject of a collaborative activity participates in the work of transforming an object into a result. In this case, the first step to model the actor rationale is creating a task, inside each actor area, to represent their participation in each activity.

Guideline 5 – Tasks representing Activities: For each subject X of activity A, a task is generated in X area with the same name given to the activity preceded by the word "Participation in … ". If the activity has two or more subjects, we generate a *Task* in each subject limited area. The *Tasks* are created to satisfy a dependence relation defined in the *Strategic Dependence Model* (see Guidelines 5.1 and 5.2.).

- **Guideline 5.1 – Dependencies between Subjects in a Single Activity:** the task representing the activity of two subjects X and Y is associated to the dependence relation generated by the guideline 2.
- **Guideline 5.2 – Dependencies between Subjects Across Activities:** In case there is a subject X in activity A which depends on something (tool, social rule or object) provided by subject Y in activity B, the tasks that represent activities A and B are associated to the dependence relation generated by the guideline 4.

Following guideline 5, the subjects of figure 2 receive in their bounded area a task representing their participation in each activity they are engaged. The lecturer, for example, receives the tasks of Participation in Construction of Project Proposals, Participation in Project Formation, Participation in Project Development and Participation in Project Evaluation. The lecturer and the students act together in the activity of Project Formation, therefore, following guideline 5.1, the task of Participation in Project Formation presented in both subject areas are connected to the goal dependence relation of Project Formation. On the other hand, the Project Formation, which has the students as one of their subjects, depends on the project proposals produced by the teacher in the activity of Construction of Project Proposals. Consequently, according to guideline 5.2, the students tasks of Participation in Project Formation is associated to the goal relation of Obtain Project Proposals

The Activity Theory provides the concepts of actions, sub-actions and operations to describe the internal behavior of the subjects in order to achieve activity success. The guidelines bellow explain how to represent the Activity Theory actions, sub-actions and operations in the Strategic Rationale Model.

Guideline 6 – Representing Actions: the actions of subject X in activity A are transformed into sub-tasks (decomposition link) of the task representing the activity A in the bounded area of X. But, if the activity has two or more subjects, then each subject receives in his limited area only the tasks representing the actions of its responsibility. Therefore, it is necessary to observe the labor division and the actions descriptions to analyze the subject responsibility in each actions.

Analyzing the labor division and the action descriptions of the Project Development activity in figure 2, for an example, it is possible to infer the lecturer actions are Student Motivations and Advise Student; and the student actions are Plan Definition, Interviews, Requirement Documentation, System Prototype and Project Presentations. To generate the organizational model of figure 4, these lecturer actions are transformed into the sub-tasks of the Participation in Project Development task in the lecturer area. On the other hand, the students actions are represented by the sub-tasks of the Participation in Project Formation task presented in the students bounded area.

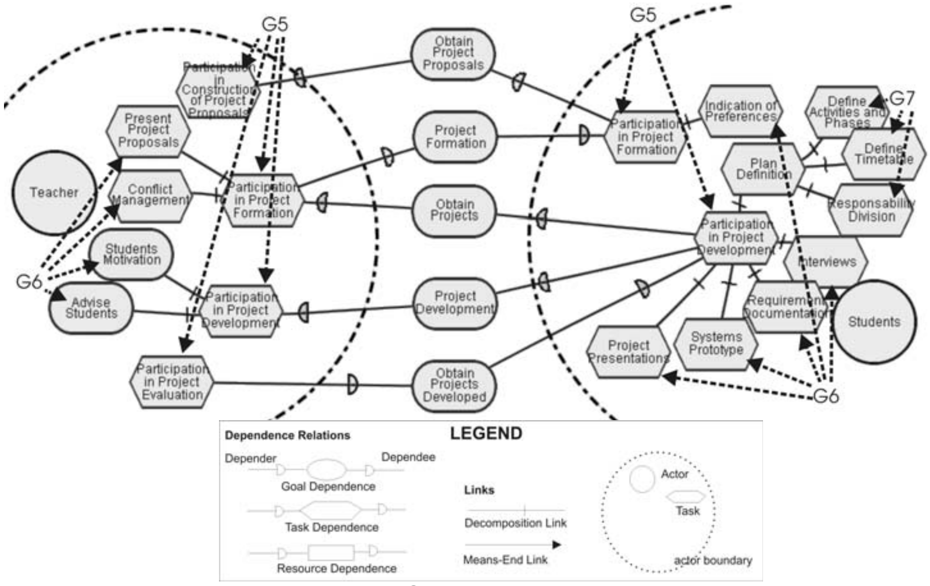

Figure 4: Strategic Rationale Model

Guideline 7 – Representing Sub-Actions and Operations: the sub-actions and operations of a given action (or sub-action) A are transformed into sub-tasks of the task representing the action (or sub-action) A by way of a decomposition link.

> *The sub-actions of the* "Students define project plan" *action (see table 2) in the Project Development activity are transformed into the sub-tasks Define Activity and Phases, Define Timetable and Responsibility Division of the task Plan Definition presented in the students area of figure 4.*

6. CONCLUSIONS AND FUTURE WORKS

Present requirement engineering approaches usually start with an empirical analysis of contextual factors in order to gradually develop requirement specifications. From our point of view, Activity Theory can provide a complementary approach starting with an abstract theoretical representation of context (obtained from ethnographical studies), and then using it to complement and guide the requirement specification using current modeling methods.

This work defines a set of guidelines to derive i* organizational models from Activity Theory diagrams. The guidelines described can be easily applied during the TROPOS [3] Early Requirement phase using a systematic analysis of the dependence relations between activity subjects, as we could verify in the context modeling of a project based learning system.

Observing the results of the case study, we can conclude that Activity Theory diagrams can be used not just as a starting point to guide the generation of the i* organizational models presented in TROPOS methodology, but also as a complementary document for a better understanding of the context being modeled. In this case, the guidelines, presented in section 5, can be used as the association links between the i* organizational elements and the detailed (context-based) Activity Theory elements.

As future work, we intend to pursue the following research topics:

- Elaborate a method that uses Activity Theory tension analysis [4, 6, 11, 18] to guide the design of new systems in TROPOS Late Requirement phase. Tension analyses are the traditional way Activity Theory researchers study and contribute to human practices development.
- In the i* organizational framework [20], a task can be decomposed in alternative sub-tasks. This alternative sub-tasks (specified by way of means-end links) indicate the task can be achieved by the conclusion of one of its subtasks. However, the Activity Theory does not allow the specification of alternative actions and operations. We intend, as a future research work, to introduce alternative actions and operations in the Activity Theory framework in order to improve its descriptive power.

7. REFERENCES

[1] S. Bødker, "Activity theory as a challenge to systems design", In H-E. Nissen, H. K. Klein and R. Hirscheim (eds.): Information Systems Research: Contemporary Approaches and Emergent Traditions, Elsevier, Amsterdam, 1991, pp. 551-564.

[2] J.B. Castro, F.M.R. Alencar, and G.A.A.Cysneiro Filho, "Integrating Organizational Requirements and Object Oriented Modeling", Proceeding of the 5th IEEE International Symposium on Requirements Engineering, Toronto, Canada, 2001, p.146 – 153.

[3] J.B. Castro, M. Kolp, and J. Mylopoulos, "Towards Requirements-Driven Information Systems Engineering: The Tropos Project", Information Systems, Elsevier,

Amsterdam, The Netherlands, The Netherlands. v. 27, n. 6, p. 365-389, 2002.
[4] P. Collins, S. Shukla, and D. Redmiles, "Activity Theory and System Design: A View from the Trenches", Nardi, B. A., Redmile, D., (eds), Activity Theory and the practice of Design. Computer Supported Cooperative Work., 2002, Vol. 11, Nos. 1-2. pp 55-80.
[5] G.G. Cruz Neto, A.S. Gomes, P. Tedesco, "Aliando Teoria da Atividade e TROPOS na Elicitação de Ambientes Colaborativos de Aprendizagem" (in Portuguese), Proceeding of the VI Workshop on Requirements Engineering (WER 2003), Piracicaba, Brasil, 2003, pp 63-77.
[6] Y. Engeström, Learning by Expanding, Helsinki: Orienta-Konsultit, 1987.
[7] M. Korpela, H.A. Soriyan, K.C. Olufokunbi, "Activity Analysis as a Method for Information System Development: General introduction and experiments from Nigeria and Finland", Scandinavian Journal of Information System, 2000, vol 12, pp 191-210
[8] A. N. Leont´ev, "The Problem of Activity in Psychology", J. Werstch (Ed.), The concept of Activity in Soviet Psychology. Armonk, New York: M. E. Sharpe, inc., 1979.
[9] C. Macaulay, D. Benyon, and A. Crerar, "Ethnography, Theory and System Design: From Intution to Insight", Journal of Human Computer Studies, 2000, 53, 35-60.
[10] L. E. G. Martins, and B. M. Daltrini, "An Approach to Software Requirement Elicitation Using Percepts from Activity Theory", Proceeding of the 14th IEEE International Conference on Automated Software Engineering (ASE´99), Florida, USA, 1999.
[11] D. Mwanza, "Where Theory meets Practice: A Case for Activity Theory based Methodolgy to guide Computer System Design", Interact'2001, Japan, 2001.
[12] B.A. Nardi, (ed), Activity Theory and Human-Computer Interaction, London: MIT Press, 1996.
[13] B. A. Nardi, and D. Redmile, (eds), Activity Theory and the practice of Design, Computer Supported Cooperative Work., 2002, Vol. 11, Nos. 1-2
[14] T.J. Richards, and L. Richards, "Using Computers in Qualitative Research". In Denzin, N. K. and Lincoln, Y. S. (eds.): Collecting and Interpreting Qualitative Materials. Sage Publication, 1998, pp. 211-243.
[15] J. Rumbaugh, I. Jacobson, and G. Booch, The Unified Modeling Language Reference Manual, Addison Wesley, 1999.
[16] V.F. Santander, and J.F. Castro, "Deriving Use Cases from Organizational Modeling", IEEE Joint International Requirements Enginnering Conference, RE´2002, University of Essen, Germany, September 9-13, 2002, pp. 32-39.
[17] C. R. B. Souza, "Interpreting Activity Theory as a Software Engineering Methodology"., Proceedings of the European Conference on Computer Supported Cooperative Work (ECSCW), 2003.
[18] P. Turner, and S. Turner, "From description to requirements: an activity theoretic perspective", Proceedings of the international ACM SIGGROUP conference on supporting group work. Arizona, 1999, pp. 286-295.
[19] S. Viller, and I. Sommerville, "Social Analysis in the Requirement Enginnering Process: From Etnography to Method", 4th IEEE International Symposium on Requirement Engineering. Limerick, Ireland, IEEE Computer Society Press, Los Alamitos, 1999, pp. 6-13.
[20] E. Yu, Modelling Strategic Relationships for Process Reengineering, Phd thesis, University of Toronto, Department of Computer Science, 1995.