# Relatório Técnico

Núcleo de Computação Eletrônica

## Cooperation Model for Learning: A System of Patterns

Flávia Maria Santoro
Marcos R. S. Borges
Neide dos Santos

NCE - 32/99

Universidade Federal do Rio de Janeiro

# Cooperation Model for Learning: A System of Patterns

**Flávia Maria Santoro**
flavia@cos.ufrj.br
COPPE/Sistemas
**UFRJ**
PO Box 68 511
Rio de Janeiro – 21 945 270
Brazil

**Marcos R. S. Borges**
mborges@nce.ufrj.br
NCE & IM
**UFRJ**
PO Box 2324
Rio de Janeiro – 20001-970
Brazil

**Neide dos Santos**
neide@cos.ufrj.br
COPPE/Sistemas
**UFRJ**
PO Box 68 511
Rio de Janeiro – 21 945 270
Brazil

## Abstract

Learning is frequently assumed as a natural result of projects performed by groups, and cooperative work tools disposal within a computational environment is enough to motivate or to induce cooperation among participants. However, many times cooperation simply does not happen within the environments, or it has to be externally articulated by a tutor. We claim that CSCL environments should be built under an explicit cooperation model to be efficient. The goal of this work is to present a Cooperation Model for Learning which is described through patterns, and to contextualize it within a software engineering approach to facilitate the development of environments aimed at generating an effective cooperation learning process.

**Keywords**: Cooperation Model for Learning, Patterns, CSCL Environments Development

# 1. Introduction

Learning is frequently assumed as a natural product of projects performed by groups, and the availability of cooperative work tools within a computational environment is enough to motivate or to induce cooperation among participants. However, in many situations the cooperation either is non-existent or needs to be promoted out of the environment by a tutor. Even in these cases, the role that the teacher should carry out to guarantee cooperation is not explicit.

Some authors, such as Roussos (1997), mention problems or unexpected results from experiences with the use of computer supported cooperative learning environments. It is also found in literature many proposals that have not been proved yet, or do not present an explicit cooperative process model to clarify how they would operate.

Santoro et al. (1999) identify the following reasons to the problem of low cooperation level within CSCL environments:

- **Culture** - one of the difficulties that determine negative results in the use of CSCL environments is that people have not been educated to work in-group;
- **Stimulus** - many CSCL environments exclusively offer tools to support group tasks execution, instead of mechanisms that favor group functions such as cognitive activity, support to individuals and well being of the group;
- **Context** – CSCL environments usually have a specific educational objective or an organizational practice training goal, but besides it, they must be integrated within other activities on which students participate;
- **Technology** - there is no tool integration within environments, and in general, people have difficulties to deal with it.

The goal of this paper is to describe a Cooperation Model for Learning and to introduce it as a basis for a software development process, providing a software engineering approach to CSCL development. We claim that CSCL environments must address issues related to the problems described (Culture, Stimulus, Context and Technology) in order to promote learning in a real cooperative way. So, we understand the necessity to provide a Cooperation Model specific for Learning, where solutions for identified problems should be contemplated.

Our Model is composed by a set of elements, which characterizes most CSCL study areas. For each area, the model includes descriptions of problems and their corresponding solutions in the form of patterns. Patterns have been used in several application areas, such as software engineering and organizational processes and they are a good way of describing solutions to problems. The model, then will work as a repository of problems and solutions patterns found in the literature or specifically developed for the area.

The resulting patterns are interrelated, forming a system of patterns, which aims at addressing several CSCL issues. The system of patterns can be used as basis for CSCL environment development, which we expect to present good results.

The rest of this paper is divided in 4 sections. In Section 2, the Cooperation Model for Learning is described. In Section 3, it is shown how the model is inserted in an also proposed development process. In Section 4, related works are presented. Section 5 concludes the paper.

## 2. Cooperation Model for Learning

Kumar (1996) emphasizes that one of the most important fields for research in CSCL is the cooperative activity modeling. According to him, modeling should involve among other things, the representation of cooperative actions, such as conflicts and resolutions; cooperative pairs roles, that could be modeled according to characteristics and deepens of cooperation process. Brna (1998) affirms that cooperation models should be explicitly defined because computational support for cooperation becomes improved by an increase of conscience of the cooperation model being supported.

This work is concerned about a representation of most CSCL issues in such a way to make it easier for someone who wants to develop an environment does not skip important cooperative process relationships, for instance, cultural aspects.

Regarding this goal - to facilitate development - the model is described through data elements. Each of these elements represents one CSCL study area. There are actually relationships among all these areas, and all of them are expanded in deeper levels of details until they reach a granularity that enables the representation through patterns.

According to Johnson (1997) patterns are attempts to describe a problem to be solved, a solution, and the context in which it is applied. A pattern captures, structures and presents key information about a domain, which specialists know how and when works. Patterns have been used in several application areas, such as software engineering and organizational processes and they are a good way of describing solutions to problems. Riehle and Züllighoven (1996) define 3 types of software patterns: (a) Conceptual Patterns: they are standard whose forms are described through terms and concepts of an application domain, they are set in metaphors and restricted to an application domain; (b) Design Patterns: they are standard whose forms are described through constructions of software projects; and (c) Programming Patterns: they are standard whose forms are described through constructions of programming languages.

Based on these definitions we can assume that the patterns of the Cooperation Model for Learning are interrelated, forming a system of patterns, and also that they are on Conceptual Patterns level, establishing an analogy with Software Engineering Analysis Phase. The system of patterns is developed from an analysis of several environments described in literature, accomplished under the optics of the Conceptual Frame for Analysis of Computer Supported Cooperative Learning Environments proposed by Santoro et al. (1999b). In this conceptual frame, dimensions and characteristics to be observed in CSCL works are defined; and guidelines for research and development are supplied. So it allows CSCL works classification and their deficiencies and qualities verification. Now a description of the model components will be made.

Several aspects are involved in a Cooperation Model for Learning. All of them are linked and attempt to produce an effective cooperative process as part of the environment. The Model proposed in this work is composed of a set of elements, which are themselves concerned to problems described through a system of patterns.

The core of this model is the **objective** and the **activity** proposed within an environment. These two elements will determine all the characteristics that the CSCL environment should have. The objective of the proposal is to bring all the issues related to group context and

culture, which will determine how the activity should be implemented in order to stimulate cooperation. In other words how the cooperative process is going to happen, who will act in this process, and what has to be stored for future utility. So, the first level of granularity is presented in Figure 1, where the issues of Context, Culture and Stimulus are addressed:

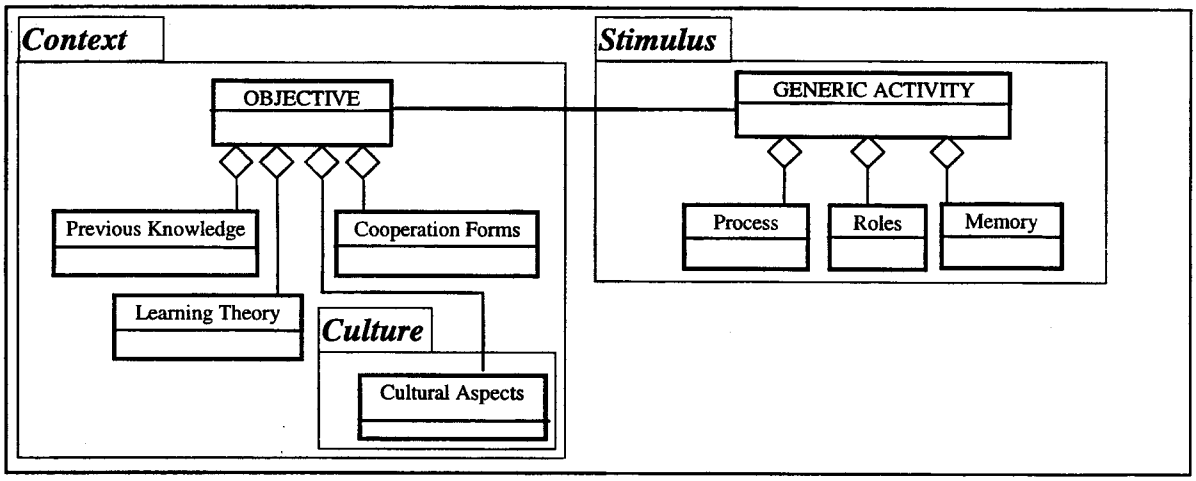


Figure 1 – Generic Cooperation Model for Learning

The Cooperation Model is based on the **Objective** that one wants to reach, for example, learning a certain concept; and in the **Activity,** that is related to the cooperative actions, for example, cooperative edition of a text. The **Objective** is related to four aspects and addresses two of the mentioned problems: **Context** and **Culture**. These aspects are:

- **Previous Knowledge** - background knowledge representation of the group.
- **Learning Theories** - a series of learning theories should form a base for the learning environment; the proposal must follow one of these theories.
- **Cooperation Forms** - the forms, which the group works, should identify and define in agreement with the objective that one wants to reach.
- **Cultural Aspects** - the cultural factors, which determine the context the group is inserted, will influence the cooperative proposal objective.

The **Activity** element is related to three issues and is linked to the problem of **Stimulus**:

- **Process** – activities to be developed by the group.
- **Roles** –functions that group members can assume, which can be different according to process diverse stages.
- **Memory** - storage of everything related to the way activities happen.

Now deeper level elements of the model, which correspond to expansions of the first level, are described.

### *2.1. Process*

The Cooperative Process is the element that describes "what people is going to do" within the environment. So, its components must define everything related to the flow of work, like tasks to be performed, how to know if they are being rightly done, and mechanisms to support the work.

A good flow of work definition, in which interdependencies are established, allied with mechanisms to support work in-group, guarantee that cooperation process will be stimulated.

The Cooperative **Process** is divided in 4 components: Action, Coordination, Evaluation, and Awareness. **Action** refers to the learning activities proposed within an environment, which will be developed by participants during cooperative learning process. These activities can be **Cooperative** or **Individual**. In fact, there will be moments for group work, and other ones for individual work. Actions can generate **Products** as a result. Besides the activity description itself, cooperative actions own a special **Dynamics**, because they are concerned with people interaction, which involve **Protocols** for each situation, and objects **Sharing**.

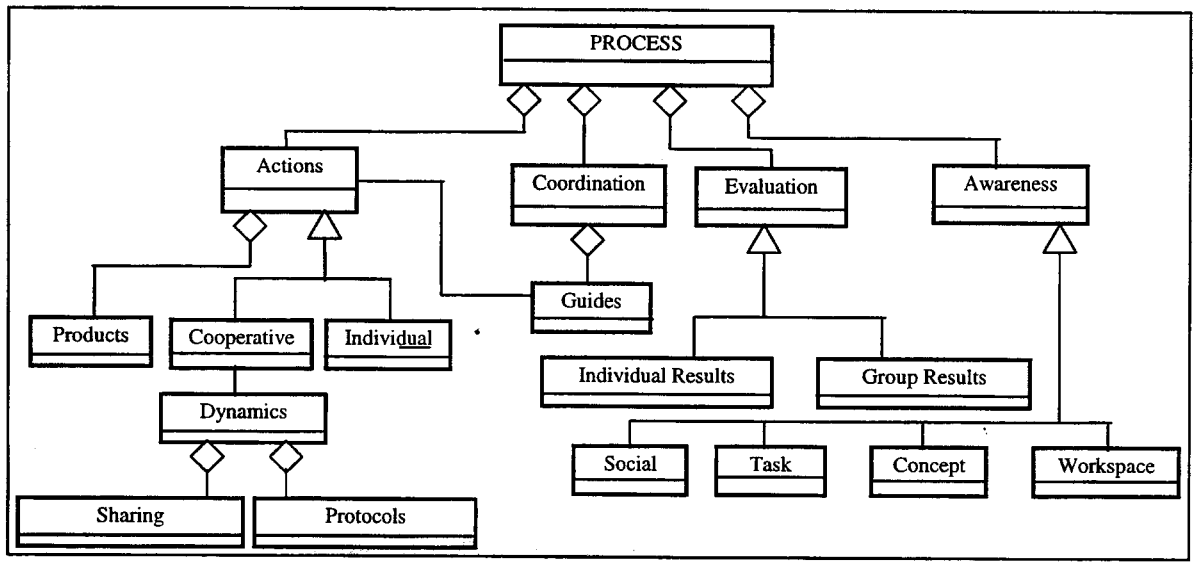


Figure 2 – Expansion of the element *Process*

**Coordination** is related to process controls and help to the apprentice. This is done using **Guides** or agents, that should monitor accomplished Actions and supply help or procedure indications. **Evaluation** is composed of important mechanisms, that measure whether objectives are being reached or not, and because of this it should be done during the whole process. In a learning process it is expected that each individual reaches a learning level of some content, and therefore **Individual Results** should be observed. Besides, in a cooperative learning process, it is also expected that the group as a whole generates some results, and therefore mechanisms to observe **Group Results** should be disposed.

**Awareness** is the element responsible to guarantee that people understand, and are conscious of the process, and of participant's interaction within the environment. Our model adopted Gutwin et al. (1995) proposal, which describes the necessity of four types of awareness elements in a CSCL environment: **Social**, that refers to the group social connections; **Tasks**, that refer to the way tasks will be accomplished; **Concepts**, that indicate how a part of the group detained knowledge is related to each member; and, **WorkSpace**, that supplies interactions awareness within the shared environment.

### *2.2. Roles*

According to the activity proposal, it can be necessary that group members play different **Roles**. The possibility to define roles within an environment makes it possible to designate

specific tasks and to best coordinate the execution of them. It is also important that people can assume different roles during the process in order to try different responsibilities. The model should make a description of the different possible roles, provide ways to nominate roles, and define how these roles will stand along the process.

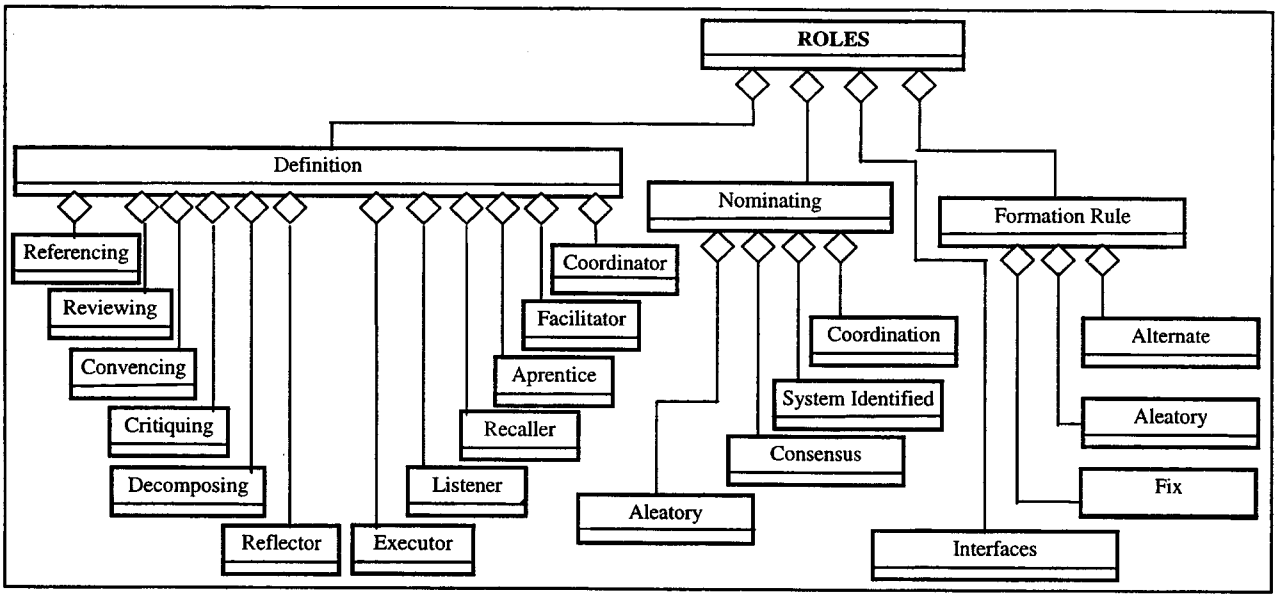


Figure 3 – Expansion of the element *Roles*

Several works (Blaye et al., 1991; McCalla, 1990) refer to different roles that could be defined in a CSCL environment. We used some of these work's definitions in our model, and add other important roles that could be defined within a CSCL environment. The element **Role Definition** is described here and provides the possible roles in such a CSCL scenario: **Coordinator** –responsible for coordinating the execution of a certain task; **Facilitator** –provides mechanisms to help the others, or to promote cooperation process; **Apprentice** - learner of a content; **Recaller**, **Listener** -pairs that are alternated as exhibitors and receivers of a content; **Executor**, **Reflector** -the individual that are more able to execute tasks, assume the executioners' role, while the shyest, assume observers' role and may produce group memory; **Decomposing**, **Critiquing**, **Convincing**, **Reviewing**, **Referencing**- several roles are carried out by group members, according to cooperative task phase.

Besides the definition of the possible roles assumed by group members during a cooperative process, it should be decided how these roles would be nominated to the individuals. **Role Nomination** can be made in several ways: **Coordination** - a coordinator decides who executes each role; **Aleatory** - members are distributed aleatorily; **Consensus** - members make a formal agreement of who will assume each role; **System Identification**- system is responsible for identifying each member role through its personal characteristics.

The designated roles can be fixed for the entire d ration of the process or not. The group can choose one of the following **Formations Rules**: **Fix** - to maintain the same roles until the end of the work; **Alternate** - to alternate roles among group members (for example, each member can act as coordinator of a task); **Aleatory** - to alternate roles among members aleatorialy. In addition to designating roles, it is necessary to implement adapted **Interfaces** appropriated to each role, and to supply necessary awareness mechanisms to the various tasks performed by the group.

### *2.3. Memory*

The cooperative activity development should be traced and stored. It is very important to preserve not only the products generated during an interaction, but the way it took place. To save the memory of an activity might be very useful as it disposes produced material and shows how the process of cooperation happened to other groups.

The **Memory** of the activity involves the following aspects: Elements, Capture, Storage and Retrieval. These issues are dependent to each other, as shown in Figure 4. **Elements** are objects related to the Activity that one wants to make persistent. They could be **Dialogues**, **Discussions**, and **Documents**, independent of their format. These elements need to be captured someway during the process. **Capture** can be accomplished in three ways: **Automatic** (the system captures the elements in a transparent way for the user); **Semiautomatic** (a part of the elements is captured automatically and other not); and **Manual** (the user should indicate to the system which elements and when to accomplish data capture). The **Storage** of captured data is related to the **Dispositive** where this will be made, **Techniques** (forms of accomplishing storage), and **Format** (data format).

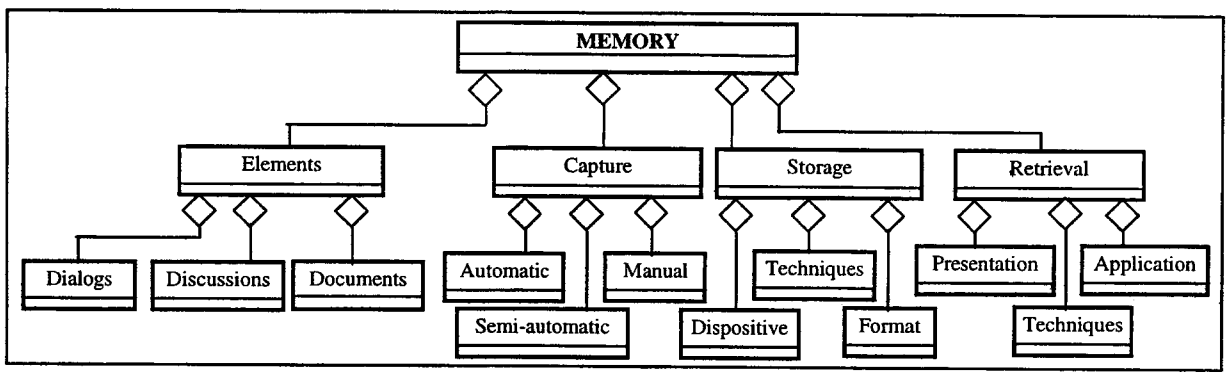


Figure 4 – Expansion of the element *Memory*

The stored Elements can be recovered in different occasions for several purposes. **Retrieval** of the elements involves: **Techniques** used to do it, that will depend on the way and format they were stored; **Presentation**, that tells how the data will be disposed for the user; and **Application**, that is related to desired objectives of cooperative process objects visualization.

### *2.4. Previous Knowledge*

A learning environment will propitiate a specific content knowledge gain. Even so, this content appropriation will only be possible depending on previous knowledge that people have. This way, it is very important to provide support to identification of group members' background.

Considering the aspect of the **Previous Knowledge** made, two analyses can be the individual and the group (Figure 5).

Each individual brings an experience or an acquired set of concepts given by formal education and daily life. This "background" can be measured or appraised under appropriate

methods. This issue is being approached by **Previous Knowledge Evaluation**. **Pre-tests** can be ways of accomplishing this evaluation.

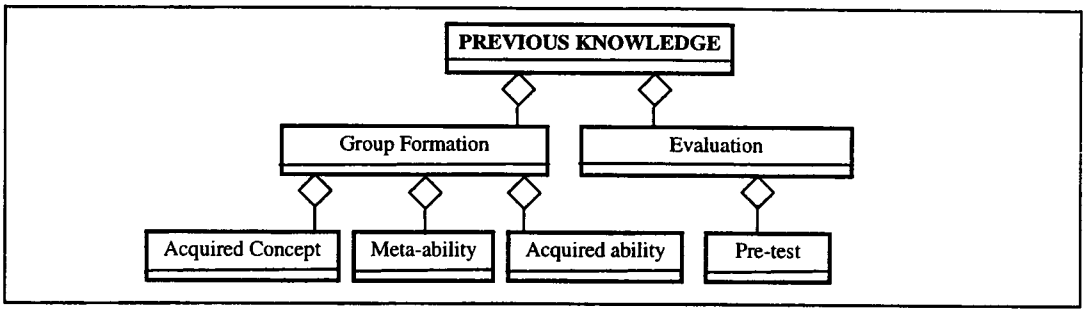


Figure 5 – Expansion of the element *Previous Knowledge*

The group as an entity also owns characteristics that evidence its knowledge background. **Group Formation**, which refers to scholarship level, group interest areas (it can be multidisciplinary), and group homogeneity can configure three knowledge types that the group as a whole possesses: **Acquired Concepts** - certain themes specific contents; **Acquired Ability** - abilities to accomplish some task type (e.g., to work with a certain computational program); **Meta-ability** - requirements or basic tools that one should know so that it is possible to reach a new learning level (for example, to have algorithm notions in order to learn how to program).

These types of knowledge should be measured in the case of individuals and verified in the case of the group. The concepts/abilities can be distributed by the members of a heterogeneous group and can be related to the roles carried out in work stages.

### *2.5. Learning Theory*

A cooperative learning proposal should be based on a learning theory that supports social interaction elements. **Learning Theory** should supply subsidies for composition and structures of learning activities, indicating how interaction can be accomplished, what should be avoided, and what is or not important in this process. There are many leaning theories that mentions interaction and cooperation as means to induce learning (Santoro et al., 1998).

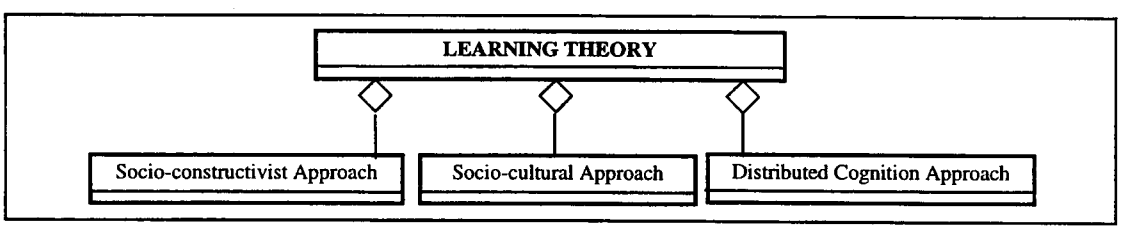


Figure 6 – Expansion of the element *Learning Theory*

To represent them in this model, we chose Dillenbourg's work that indicates three basic approaches (Figure 6). Dillenbourg et al. (1994) state that there are three different theoretical approaches able to understand the nature of cooperative learning: either focusing on the individual's vision (independent cognitive systems that interact), or on the group (cognitive system with own characteristics): **Socio-Constructivist Approach-** understands

individual's cognitive development as a result of a spiral causality, where a given development level allows participation in certain social interactions, which produce new individual states, and because of this allow more sophisticated social interactions and so on; **Socio-Cultural Approach-** is founded in the theory developed by Vygotsky and focused in causality relationship between social interaction and cognitive changes in the individual; **Shared Cognition Approach:** . focused in social context, where the environment, which includes a physical context and a social context, is integral part of the cognitive activity, and not merely a group of circumstances in which independent cognitive processes happens.

### *2.6. Cooperation Forms*

A Cooperative learning proposal can be centered in different forms or models. This is related to level and engagement in the cooperation process. Depending on how the group has been working or how a new proposal is made, a work configuration can be defined.

Brna (1998) describes some of these models and affirms that being conscious of which one is used can improve computational support. In the context of this work six **Cooperation Forms** (Figure 7) will be considered, and each one owns a cooperation level (lower or higher): **Work Division** – in this case a division of tasks is made, and each member of the group is responsible for a task; **Cooperation State** – there are individual and in-group work moments, but the group is sharing a cooperative state; **Cooperation as Final Purpose** - the work objective is to learn how to cooperate; **Cooperation as Means** - the work objective is to learn something, using cooperative techniques; **Formal Cooperation** – group members make an agreement to accomplish a work cooperatively; **Informal Cooperation** - group members work cooperatively, but there is not a formal agreement, the cooperation happens spontaneously.

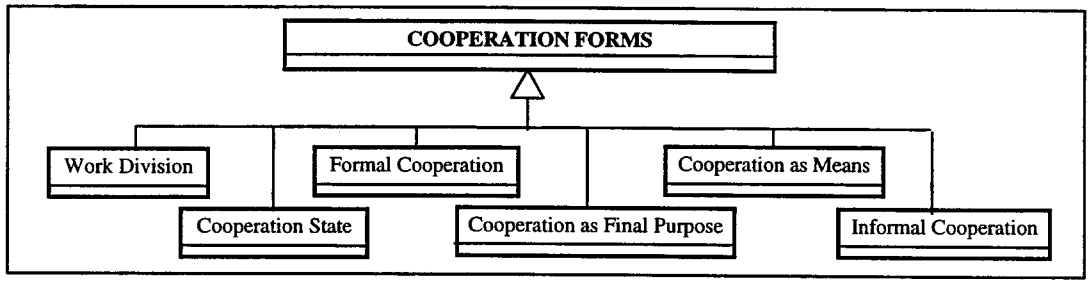


Figure 7 – Expansion of the element *Cooperation Forms*

These forms (or their composition) are choices done by the group or imposed by of task proposal type, and should be related to learning theory and cultural factors.

### *2.7. Cultural Factors*

A proposal that involves interaction among people cannot avoid considering cultural factors related to context where the group is inserted. Some problems that may arise within the group could be predicted if the culture was previously identified. Besides this issue, the group can be motivated according to its necessities and expectations. The environment can develop ways to integrate people using common communication patterns.

According to social sciences researchers (McGrath, 1993) and works on organizational culture (Robbins, 1994), a set of **Cultural Aspects** (Figure 8) specifically for cooperative learning environments was specified.

These factors are divided in **Individual Properties** (each person owns individual components that will influence group behavior), and **Group Structure** (that will demonstrate how the group is organized, and how people are situated inside of it). Among the Individual Properties, **Characteristics**, **Beliefs**, **Habits**, **Age**, **Sex**, **Needs**, **Expectations**, and **Motivations** are met.

Group Structure understands: Disposition, Motivation and Relationship. **Disposition** indicates how individuals are distributed inside of the group, and involves **Composition** (hierarchy); **Work Division** (division of tasks among people); and **Power** (description of power structures inside the group).

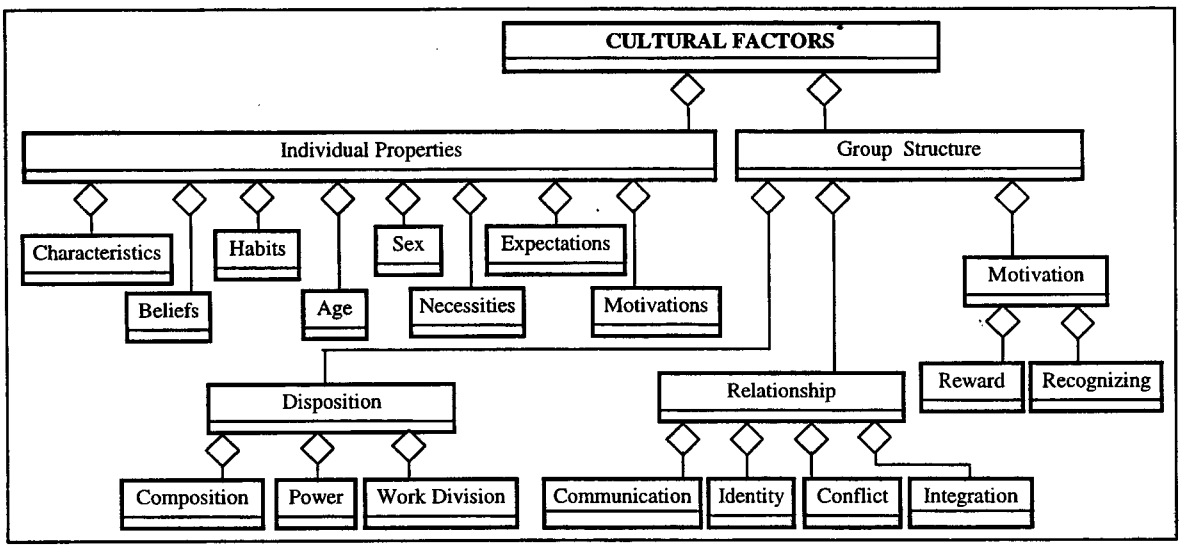


Figure 8 – Expansion of the element *Cultural Aspects*

**Relationship** describes possible ways used by individuals to interact within the group and is resumed in **Communication** - set of patterns adopted by the group to communicate, for example, the degree in which the communication in an organization is restricted to the formal hierarchical line; **Integration** - degree in which group members are encouraged to operate in a coordinated way; **Identity** - degree with which group members identify themselves with the organization; **Conflict** - way how conflict treatment is foreseen in cooperative environment. **Motivation** embraces **Reward**, when the environment works with reward allocation; and **Recognition**, that is the form how incentive is provided to group member's participation, through recognition of their acting.

### *2.8. Description of the Model through Patterns*

For each element of the model described the most important problems are identified and described as patterns, like the example bellow Pattern Workspace Awareness in CSCL (Figure 9), related to the element Process-Awareness-Workspace (in Figure 2).

The set of all conceptual patterns, such as this one, form the system of conceptual patterns. The next stage is to transform conceptual patterns into design patterns. Some patterns of the

system may be directly transformed, while others will serve just as a guideline to the implementation. As an example for the first case, the conceptual pattern of Figure 9, will have an associated design pattern representing the software design of the solution. In the second case, a pattern related to learning theory approach will only guide how to deal with evaluation techniques.

**Pattern Name**: WorkSpace Awareness in CSCL

**Problem**: What elements should be disposed in a CSCL environment in order to guarantee workspace awareness among the members of the group?

**Context**: In a CSCL environment, awareness of participants is one of the key elements to propitiate an effective interaction. Knowing what someone is doing in a certain moment can make people interact easily. The access to information on contributions and tasks already completed is also an important factor, because it can approach people with common interests.

**Forces**: Workspace awareness reduces the overhead of the work in-group, allowing a more natural and effective interaction. Workspace awareness facilitates students' practices that allow cooperative learning to occur.

**Solution**: The environment should provide a representation of each member of the group within the workspace, so that the whole group can visualize "where he is", "what he is doing ", and "what he had already done ". This representation can be graphic, iconic, through windows, or virtual reality.

**Related patterns**: Social Awareness in CSCL, Task Awareness in CSCL

**Known uses**: CSILE implements workspace awareness through structured message notion, where an author is represented by its initials, and therefore other members get to know what he is doing or have already done. Following the same line, CSILE, Collaboratory Notebook and CaMILE also implement structured messages where the authors are identified for its names.
In NICE (Roussos, 19997), the participants of a work section are represented by avatars and everybody can visualize their movements and actions inside of the workspace. CLARE supplies a vision of all the works accomplished by group members in a specific window.

Figure 9 – A Pattern Example

## 3. A CSCL Development Process

To contextualize the use of the proposed model, we present a Cooperative Learning Environments Development Process, which is characterized by the following steps, according to Santoro et al. (1999a):

1. Development of a **Generic Cooperation Model for Learning** described through Patterns, that suggests integrated solutions for elicited problems;
2. Description of a specific **Domain of Tasks** in CSCL, which could be for example, Project Development;
3. Definition of a **Technology** that should minimize issues related to integration and interface;
4. Design of a software **Framework**, which is based on the model proposed and generates a Framework Arquitecture;
5. Implementation of the Framework, selecting a programming language for it;
6. Definition of a **Theme**, which is concerned to a content dependent to where it will be applied;
7. Instantiation of the Framework to the specific Theme, producing different **Computer Supported Cooperative Learning Environments**, which incorporate components of an explicit cooperation model and for this reason should produce better results.

According to Johnson (1997), a framework can be defined as a reusable project of a system or parts of a system, represented by an abstract group of classes and the way its instances interact. In summary, the Cooperation Model should facilitate the development of a framework that incorporates a series of descriptions and definitions on the cooperation process that one wants to stimulate. That framework should make it easier to the software developer to implement better environments.

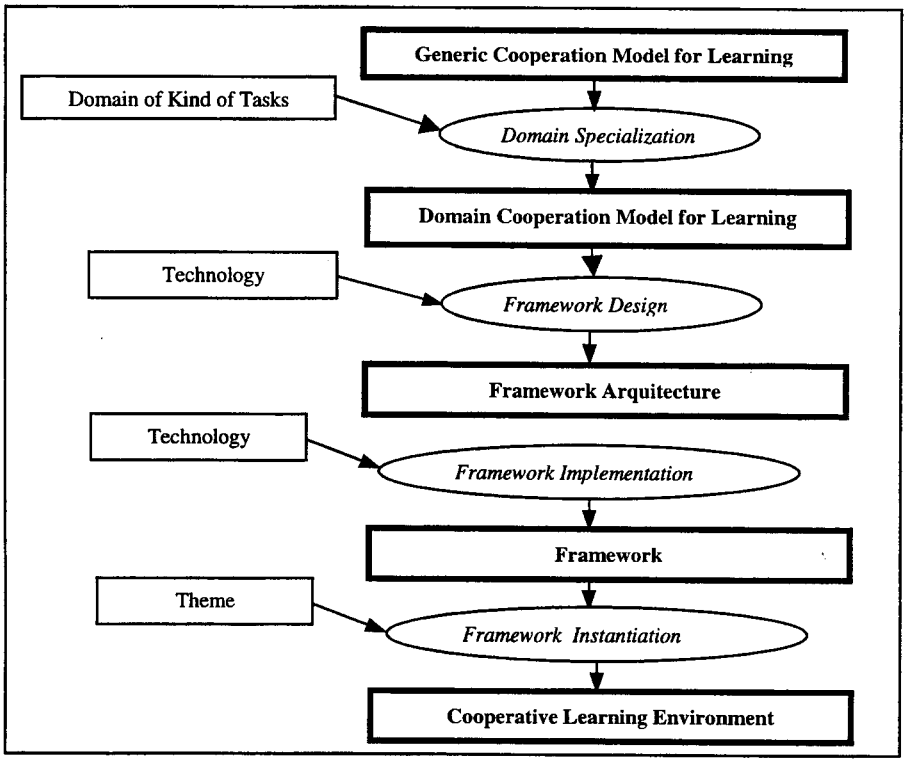


Figure 10 – Development Process based on the proposed Model

## 4. Related Works

In the literature we can find some works on groupware for learning development. The article "Cooperation Model for Teaching/Learning of Modeling Disciplines" by Becker and Zanella (1998) introduces a structured cooperation model to aid the teaching/learning process of modeling disciplines through development of exercises, criticism and discussion of them. It provides a general framework that defines: (a) a process; (b) roles; and (c) shared objects. Becker and Zanella's work is restricted to a domain of applications (modeling) and to a specific type of processes. Their model only considers some elements of the process, but not other issues pointed out in our proposal, such as cultural aspects and learning theories.

SECAI Model is defined within CLARE Environment, which is a distributed learning environment which goal is to facilitate learning through collaborative knowledge construction (Wan and Johnson, 1994). The SECAI Model (Summarization, Evaluation, Comparison, Argumentation, and Integration) defines an explicit model of process for collaborative learning of scientific texts, that metaphorically "pulls " the apprentices of an

external, isolated and individual position to an internal perspective, integrated and collaborative. This model is also limited to represent some aspects related to the cooperation process. Our proposal goes further defining a more complete conceptual model, addressing other issues, not contemplated in SECAI, such as cultural aspects, learning theories, roles and evaluation of learning.

ARCOO project was developed with the goal to support cooperative learning in distributed environments, where interaction among individuals occurs in the search of a solution for a problem. The requirements of ARCOO project consider social, cognitive and technological aspects: group structure, creativity, planning, activities integration, group conscience, access information support, several communication modalities support, several sharing representation forms of the work objects and of knowledge. (Barros and Borges, 1995). ARCOO uses a model of objects to represent the interactions within the environment, while our proposal delivers a system of patterns to develop a reusable framework. ARCOO does not address all aspects of our proposal and is specified to remote and distributed learning.

## 5. Conclusions

Research in the area of CSCL found that many of the developed environments do not reach the intended success, that is, to promote cooperation among members of a group to improve learning. Most of the problems are related to **stimulus**, **context**, **culture** and **technological** problems. In this work we present a Cooperation Model for Learning as part of a proposal to solve some problems in the area of CSCL. The goal is to facilitate the development of cooperative learning environments, supporting the aspects mentioned previously- stimulus, context, culture and technological.

We claim that building CSCL environments under a software engineering approach, where the developer can use a safe model as basis for development, should produce effective learning by cooperation as a result. One major contribution of our proposal is to analyze non technical problems - cultural influence, educational approach, environmental context - and conjugate them to cooperative activities and issues associated to them- objects sharing, coordination, evaluation, awareness - providing guidelines for how to implement the activities with a good theoretical basis.

Comparing the related works with our proposal, we conclude that few of them are based on an explicit model, and also none of them address all the issues that we intend to support, even for a specific domain. Despite the number of computer supported cooperative learning environment proposals, the use of a system of patterns was not yet explored in CSCL, and specific frameworks practically does not exist for CSCL processes.

Our model includes a series of learning theory approaches, cooperation forms, cooperative activities, and context issues. The system of conceptual patterns obtained from the model will be translated into design patterns and then a software framework can be developed. The framework should be restricted by a set of choices made form those available in the model and define a particular domain of tasks to form the basis of an environment. The resulting environment will be used in a real situation in order to demonstrate the advantages of our approach.

## Acknowledgments

This work was partially supported by grants from the Brazilian Research Councils (CAPES and CNPq).

## References

**Barros**, L.A.; **Borges**, M.R.S. (**1995**). ARCOO- A Support System to Distributed Cooperative Learning. In Portuguese: ARCOO - Sistema de Apoio à Aprendizagem Cooperativa Distribuída. *Proceedings of VI Simpósio Brasileiro de Informática e Educação.*

**Becker**, K., **Zanella**, A.N. (**1998**). A Cooperation Model for Teaching/Learning Modeling Disciplines. *Proceedings of CRIWG'98 Cyted Ritos International Workshop on Groupware, Brasil.*

**Blaye**, A.; **Light**, P.H., **Joiner**, R., **Sheldon**, S. (**1991**). Joint Planning and Problem Solving on a Computer-Based Task. *British Journal of Development Psychology 9:471—483.*

**Brna**, P. (**1998**). Modelos de Colaboração. *Revista Brasileira de Informática e Educação, No.3, pág.9-15.*

**Dillenbourg**, P., **Baker**, M., **Blaye**, A. e **O'Malley**, C. (**1994**). The Evolution of Research on Collaborative Learning. (WWW document) http://tecfa.unige.ch/tecfa-research/lhm/ESF-Chap5.text

**Gutwin**, C., **Stark**, G., **Greenberg**, S. (**1995**). Support for Workspace Awareness in Educational Groupware. *Proceedings of Computer Supported on Collaborative Learning'95, EUA.*

**Johnson**, R.E. (**1997**). Components, Frameworks, Patterns. *ACM 0-89791-945-9/97/0005.*

**Kumar**, V.S. (**1996**). Computer-Supported Collaborative Learning: Issues for research. *8th Annual Graduate Symposium on Computer Science, University of Saskatchewan* http://www.cs.usask.ca/grads/vsk719/academic/890/project2/project2.html

**McCalla**, G. (**1990**). The Central Importance of Student Modeling to Intelligent Tutoring. *Technical Report, ARIES Laboratory, University of Saskatchewan, Saskatchewan S7N 0W0 Canada.*

**McGrath**, J. (**1993**). Groups and Human Behavior. *Readings in Groupware and Computer-Supported Cooperative Work Assisting Human-Human Collaboration* (Excerpt from Groups: Interaction and Performance, 1984, Prentice Hall), Edited by Ronald M. Baecker, Morgan Kauffman Publishers, Inc.

**Riehle**, D. e **Züllighoven**, H. (**1996**). Understanding and Using Patterns in Software Development. *Theory and Practice of Object Systems 2, 1 Page 3-13.*

**Roussos**, M. (**1997**). Issues in the Design and Evaluation of a Virtual Reality Learning Environment. *Master Thesis, University of Illinois, Chicago, USA.*

**Santoro**, F.M., **Borges**, M.R.S., **Santos**, N. (**1998**). Computer Supported Cooperative Learning Environments: A Theoretical Approach. In Portuguese: Ambientes de Aprendizagem Cooperativa Apoiados por Computador: Uma Perspectiva do Referencial Teórico. *Proceedings of Taller Internacional de Software Educativo.* Santiago, Chile.

**Santoro**, F.M. (**1999a**). Doctoral Qualifying Exam. COPPE Sistemas, Federal University of Rio de Janeiro, Brazil.

**Santoro**, F.M., **Borges**, M.R.S., **Santos**, N. (**1999b**). A Framework for Analysis of Computer Supported Cooperative Learning Environments. To be presented at Ed Media'99, Seattle, USA.

**Wan**, D., **Johnson**, P.M. (**1994**). Computer Supported Collaborative Learning Using CLARE: the Approach and Experimental Findings. *Proceedings of 1994 ACM Conference on Computer Supported Cooperative Work, Chapel Hill, North Carolina.*