Betbeder M-L., Tchounikine P., (2003), Symba a tailorable framework to support collective activites in a learning context. In CRIWG, 9th International Workshop on Groupware, 2003, Autrans (France), p. 90-98, Springer Verlag

Symba: a Tailorable Framework to Support Collective Activities in a Learning Context

Marie-Laure Betbeder[1], Pierre Tchounikine[1]

[1] LIUM , Université du Maine
Avenue Laënnec, 72085 Le Mans cedex 9, France
{Betbeder, Tchounikine}@lium.univ-lemans.fr

Abstract. We present an approach that aims at introducing tailoring capacities in a framework designed to support collective activities in a learning context: students state their organization (using a task conceptual notion modelled following Engeström's triangle) and are then proposed with a reification of the adopted organization, and, in particular, the set of tools they have asked for. This approach has a double advantage, making students achieve a reflective analysis of their activity and enabling them to tailor the activity-level environment without having to cope with a programming-like tailoring language.

1 Introduction

The general context of this work is the design of Web-based frameworks that support Collective Activities in a Learning Context (CALC). We study, through iterative experiments, what features can be introduced in a framework in order to support students' activities and make them practice (and learn how to manage) collective work.

The literature and are own previous experiments highlights that when facing collective activities (1) students experience great difficulties in structuring their organization and (2) it is not possible to predict how students will organize themselves and, consequently, the tools (e.g., Chat, Mail or file exchange tools) that they will use (and how) to achieve their activity. Therefore, a framework designed to support such CALC must support the students in organizing themselves, but it must also permit students to decide the tools that should be accessible in order to achieve the tasks they define: it must be tailorable.

In this paper we propose an approach based on the principle of delegation of tasks. The task notion is proposed to the students as a conceptual tool that enables them to describe their organization (plans as list of tasks, delegation of tasks to individuals or subgroups) and the tools they want to be accessible when achieving these tasks. Tailorability is provided by (1) the fact that students can define the tools and resources that will be accessible to them when achieving the tasks they have planned and defined and (2) the fact they can delegate some specific tasks to some software agents.

Section 2 presents the interest of tailorability in a CALC context and the general principles that we propose. Section 3 presents how we have put these principles into practice in Symba, the framework we have currently developed and experimented.

The paper is illustrated with examples from our experimental field, a computer science curriculum (5th year of University) that mixes face-to-face and distance students. Within this curriculum we consider activities[1] such as proposing to subgroups of 6-7 students to collect some information about a given subject and then to collectively construct a synthesis by confronting their points of view[2] through Symba. Such an activity mixes individual and collective phases, that can be performed synchronously or asynchronously and correspond to collaboration or cooperation (this is why we use the "collective" term).

2 Interest of Tailorability and General Principles

2.1 Tailoring the Support Proposed by a Framework within a CALC

Many works from the CSCW and CSCL communities have emphasized that groupwares rigidity is a core problem. The support required by a particular group at a particular moment depends on different context-dependent features and, in particular, the organization it has adopted. This organization, as pointed out in our experiments [3] and in the literature [14][4][5], cannot be anticipated (given an identical situation, groups adopt different organizations and, in any case, this organization generally evolves during the activity). Designing a framework that is supposed to support students must face this central challenge.

A computer system is said tailorable if it proposes its users with some means to modify itself in the context of its use, as one of its functionalities [10][11]. Tailorability, if it focuses on making the appropriate support arise and meet the students' needs [2], appears a promising approach.

2.2 Tailorability as a Means to Make Students Work out Organization Features

In the context of CALCs, providing tailorability features has a double advantage. First, it allows students to achieve the work within a framework that proposes a support adapted to their wills. Second, in order to define this support, students have to practice a reflective activity on their organization. This is of a particular interest for us as it corresponds to our core pedagogical objective, and we believe it is interesting for most CALCs situations.

We distinguish three natures of support to be provided to students, each of them associated with a kind of tool (in the sense intended by the activity theory [7]): (1) conceptual support such as the one provided by the task notion, a "conceptual tool" that allows students to conceptualize the work to be achieved, (2) support to achieve a task such as the one provided by "general-purpose tools" (e.g. a Mail tool or a file

[1] The "activity" term will used in this paper to denote both what students are asked to do (the "pedagogical activity") and in the broader sense of "students' activity". Additional precision will be made when the context is not clear enough to distinguish these meanings.

[2] As an example, in the last experiment students had to identify how conceptual maps can be used to describe a curriculum (step 1), propose different individual points of view (step 2) and then construct collectively, from their individual productions, a conceptual map describing their curriculum (step 3). Such activities are designed with a double pedagogical objective, making students work on some feature of the curriculum domain and making them practice collective work.

exchange tool, that can be used as means to achieve different tasks) and (3) task handling support provided by “task oriented tools”, i.e., tools that achieve specific tasks (e.g., collect different students’ productions and make them available to the group or organize a vote).

The form of tailorability we propose addresses issues (2) and (3) and is based on the following two-step principle. First, in accordance with the pedagogical objective of making them aware of the need for organization, students state their organization explicitly (the objective is not that the framework should adapt itself automatically to the students needs): they identify and describe the tasks to be achieved (e.g., objectives, actors, tools to be used, dates or input and output) and structure them as plans. They are then proposed with an activity-level that reifies the adopted organization (list of tasks descriptions) and provides the general-purpose and task-oriented tools they have asked for. As an example, within the type of activities we consider, students can state that they want to be proposed with an interface that embeds a frame to browse their individual productions, a Chat to confront their ideas synchronously and a voting tool (this last being, in our framework, a task-oriented tool).

Tailorability could be further introduced at the conceptual tools level, i.e. by proposing different possible modelling primitives or parametrizable primitives. We however consider that this level of tailoring appears too confusing in a learning context and propose (for the moment) a single primitive, the task notion, that can be linked both to Newell’s rational principle [12] and the activity theory notions [7].

2.3 Discussion

Making students explicit their organization addresses both the general objective of supporting them in their collective task and our core pedagogical objective of making them work out organizational features. This must not be understood as a desire to make students bury their organization in intangible plans. We believe it is pedagogically unsuitable (and in fact unrealistic if one adheres to Suchman’s situated action ideas [14]) to constrain students to follow exactly *a priori* plans. Plans are used here as proposed in [1], i.e., means to organize the work and reflect the responsibility of the involved actors. When achieving their work, students can follow or not the adopted organization. However, *a minima*, making students explicit their plans makes them address organizational issues and, as the organization is reified in the tool, it can serve as a basis (an intermediation role) for debriefing activities with the tutor during and after the activity. Within our approach students have however an additional reason (that they perceive directly) to work out and explicit their organization: it allows them to define what tools will be accessible to achieve their work.

3 Principles of the Approach Adopted in Symba

3.1 General Theoretical Background: the Activity Theory

We consider the overall pedagogical activity that the students are proposed with as a set of *n* “situations” in which actors have to achieve a task whilst respecting a set of rules and by the means of a set of tools. Each task is viewed as an instance of the model issued from the activity theory [7] known as the Engeström's triangle [6] (the production aimed systemic model constituted of the subject, the object, tools, community, rules and division of labour). This model and the mediation relationships it highlights [8] are the conceptual notions that the students are proposed with in order to conceptualize their organization (this prescriptive use of the Engeström's triangle is a working hypothesis).

3.2 The Task Definition / Delegation Principle

A classical problem when designing a tailorable system lies in the tailoring language. Users cannot be expected to be skilled programmers. Moreover, as the tailoring will occur while using the system, the way users can tailor their system must be related to their current task so that they don't have to leave the application domain to work on the underlying domain of programming, a shift that would cause a breakdown in the activity flow [9].

Within our context, the tailoring occurs while organizing the collective work. It must therefore be expressed in an organizational semantic. The approach we propose is to define the tailoring through a "task definition / delegation" semantic. Instead of presenting students with a predefined framework whose functionalities can be tailored, we ask them to declare their organization by specifying a plan as a succession of activity-level tasks (e.g., "discussion" or "vote") and delegating each of these tasks to an individual or a subgroup of students[3]. This is achieved through dedicated interfaces.

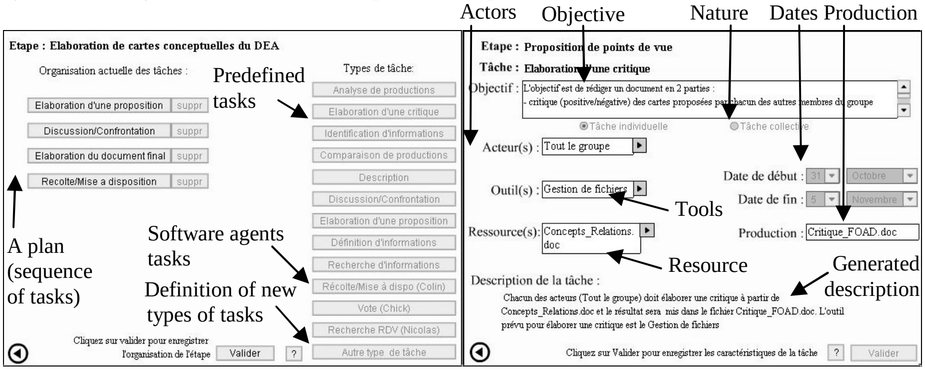
Symba's task description interface (Fig. 1, right part) denotes Engeström's triangle notions. For every task, the students have to define the objective (in natural language), the nature (individual or collective), the subject (an individual, a sub-group, all the group), the beginning and ending dates (rules), the tools (the general-purpose tools that will be accessible at the activity-level to achieve this task), the resources and production (files names). The system dynamically generates a natural language explicitation of the description in order to avoid possible misunderstandings of the interface. The description (and, therefore, the support provided by the framework) can be changed at any time (including while the task is achieved).

The students can design the overall organization as a plan (Fig. 1, left part). A plan defines how the students intend to decompose the general task they have been proposed with into subtasks and how these subtasks will be scheduled (each of these subtasks being further described with the task editor). In order to help students, the plan editor proposes a set of predefined abstract tasks types. As an example, given the type of activities we propose the students with, we propose tasks types such "search for information", "analyze production" or "discussion", that the students then customize through the task editor. The list of tasks types is to be defined according to the pedagogic context. Students can also define new types of tasks if required.

[3] From another point of view: organization is a meta-level task that consists in defining and delegating activity-level tasks. For this meta-level task the subject is the group and the tools are (1) the conceptual notion of task (students must conceptualize their activity as tasks) and (2) editors that enable students to define the plan and its tasks. As stated before, these features are not tailorable.

Let us recall that the plans we are concerned with here are (1) means to make students work out organization features (before, while and after achieving the work) and (2) resources to carry out the work [1]. A plan does not intangibly determine the students' activity, it reifies how they perceive their organization.

Both task and plan editors interfaces are shared, students can connect themselves synchronously and browse the plan / tasks while they define it (all of the students see the interface, one of them can modify it, using a turn taking system). The editors are associated with a Chat (not visualized in Fig.1) that enables students to discuss synchronously about the tasks and plan descriptions.


Etape : Elaboration de cartes conceptuelles du DEA

Organisation actuelle des tâches :

- Elaboration d'une proposition
- Discussion/Confrontation
- Elaboration du document final
- Recolte/Mise a disposition

Types de tâche:

- Analyse de productions
- Elaboration d'une critique
- Identification d'informations
- Comparaison de productions
- Description
- Discussion/Confrontation
- Elaboration d'une proposition
- Définition d'informations
- Recherche d'informations
- Récolte/Mise à dispo (Colin)
- Vote (Chick)
- Recherche RDV (Nicolas)
- Autre type de tâche

Cliquez sur valider pour enregistrer l'organisation de l'étape

Etape : Proposition de points de vue

Tâche : Elaboration d'une critique

Objectif : L'objectif est de rédiger un document en 2 parties :
- critique (positive/négative) des cartes proposées par chacun des autres membres du groupe

● Tâche individuelle ○ Tâche collective

Acteur(s) : Tout le groupe

Outil(s) : Gestion de fichiers

Ressource(s): Concepts_Relations.doc

Date de début : 31 Octobre

Date de fin : 5 Novembre

Production : Critique_FOAD.doc

Description de la tâche :
Chacun des acteurs (Tout le groupe) doit élaborer une critique à partir de Concepts_Relations.doc et le résultat sera mis dans le fichier Critique_FOAD.doc. L'outil prévu pour élaborer une critique est le Gestion de fichiers

Cliquez sur Valider pour enregistrer les caractéristiques de la tâche

A plan is described through the plan editor (left snapshot) by defining or selecting tasks from the list of predefined tasks and scheduling them. The right snapshot presents one of the tasks' definition through the task editor. Its description in natural language generated by the system is: every actor (of the group) must elaborate a criticism from the file "Concept_Relation.doc" and put the result in the file "Critique_FOAD.doc"; the tool that is asked for is a file exchange service.

Fig. 1. Plan and task editors (snapshots from the experiment)

We use this task definition / delegation principle as an unifying way to introduce the two different forms of support presented in §2.2., general-purpose tools and task-oriented tools.

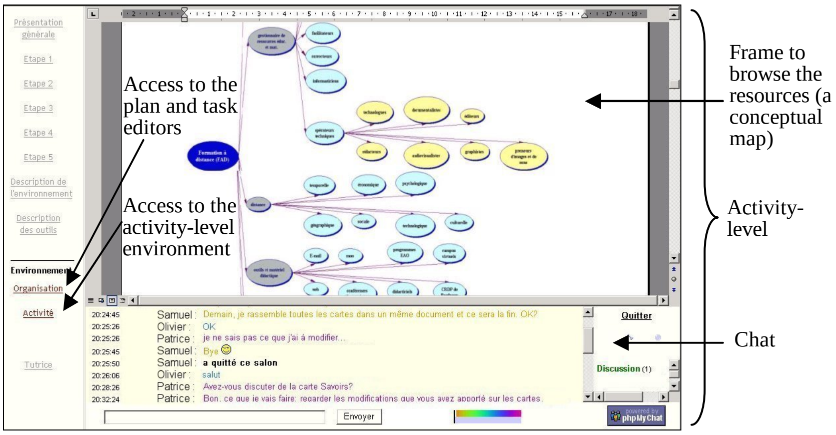
3.3 Task Definition / Delegation as a Means to Specify the Activity-Level General-Purpose Tools

When constructing a plan by instantiating and scheduling tasks types such "search for information" or "analyze production", students describe what they intend to do and how they intend to do it, i.e., what tools they want to be proposed with at the activity level. From each task description is generated a Web page that proposes an integrated access to the different general-purpose tools and resources that have been asked for (cf. Fig. 2, an activity-level generated for a task which tools are a browser and a chat). This Web page is the frame within which the students will achieve the task.

Symba does not embed specific tools, it allows an access from the framework to external tools ("integration" approach)[4]. The framework is therefore evolutive: adding

[4] Three levels of tailoring can be distinguished [10]: customization (modifying the system by choosing its attributes' values from a predefined set), integration (adding new functionalities to the system by linking predefined components together with an integration language) and extension (adding new code to the

a tool to Symba (i.e., enabling the students to use it) requires describing the tool in order to allow the students to understand its possible uses and programming a piece of software that generates an access to this tool when the students select it.



20:24:45 Samuel : Demain, je rassemble toutes les cartes dans un même document et ce sera la fin. OK?
20:25:26 Olivier : OK
20:25:26 Patrice : je ne sais pas ce que j'ai à modifier...
20:25:45 Samuel : Bye
20:25:50 Samuel : a quitté ce salon
20:26:06 Olivier : salut
20:28:26 Patrice : Avez-vous discuter de la carte Savoirs?
20:32:24 Patrice : Bon. ce que je vais faire: regarder les modifications que vous avez apporté sur les cartes.

Fig. 2. Snapshot of a generated activity-level environment

3.4 Task Definition / Delegation as a Means to Integrate Task-oriented Tools

We have introduced in §2 a distinction between “general-purpose tools” (e.g., a Mail tool) and “task oriented tools”, i.e., tools that are identified as achieving a specific task (e.g., collect different students’ productions and make them available to the group or organize a vote). Within this distinction lies a large part of CALC frameworks flexibility problem. Groupwares that only propose general-purpose tools do not constrain the students but offer them a limited support. At the other side of the spectrum, groupwares that implement complex processes support students if they adhere to the underlying organization, but constrain them if not.

Within Symba, services such as “collecting documents from the different actors” or “organizing a vote” (i.e. task-oriented tools), that are traditionally implemented as functionalities embedded in the framework, are defined as predefined tasks that can be delegated to software agents. These tasks are proposed with the others in the set of predefined abstract tasks proposed by the plan editor (Fig. 1). In other words, we provide a service to the students’ community by integrating in this community a software agent that proposes the required competence to the community rather than by a black-box tool. Within this approach, students do not use tools, they define a task to be handled by a software agent, in a similar way that they define some other tasks to be handled by human agents. The emphasis thus remains on organisational features (defining and delegating tasks), i.e., the point we want them to work out.

We have defined and implemented three software agents in the current Symba version: Nicolas can find a date for a meeting when (for example) a group prepares a

application). Components integration is a research domain in itself, and how heterogeneous software components can be integrated is the object of a large amount of studies [4]. Tools integration in Symba is of course a very light approach, but, given our objectives, sufficient.

synchronous meeting. Chick can organize a vote when the group must decide something collectively. Colin can handle document flow produced or to be used by the group. These agents communicate with the students by Chat, Email and file transfer directly (Email) or within the activity level (e.g., resources handling). Their precise definition (e.g., dates and ballots for a vote) are defined through the task editor, similarly as the tasks delegated to human agents. Due to limitation place these agents issues are not emphasized here, see [15] for details.

4 Conclusions

We have presented an approach to the tailoring of CALC frameworks where students state their organization and are then proposed with a reification of the adopted organization, and, in particular, the set of tools they have asked for. This approach has a double advantage. First, when defining the plan and its tasks, students achieve a reflective activity on the process, the objectives, the resources and products, their method for working collectively or the best fitted tools. Second, through this process, they can tailor the activity-level environment of each task. The task definition / delegation principle is used as a unifying way to tailor the support that is provided by the framework: students define what has to be done and how for all of the tasks, the environment generates an activity-level that embeds the tools that have been asked for (for general tasks) and/or provides software agents that can achieve some particular given tasks (task-oriented tools).

From the tailorability point of view, the approach we propose is general, Symba is one of different ways to put it into practice. It opens different challenging questions to be studied such as how students react to such tailoring capacities or what type of services (task-oriented tools) can be proposed as software agents (including proposing agents that perform a given task following different protocols or means). It will be interesting to propose different approaches to the description of a plan in order to analyze what types of representations are used by students and how they use them and to study (through successive experiments) the crystallization that could denote the definition and reuse of new types of tasks and/or of plan patterns.

From a pedagogical point of view, we believe that Symba can be used as an integrative framework for collective activities when supporting organization and/or providing tailorability features can be of additional value. For instance, Symba is a possible way to operationalize collective activities designed from the Tecfa seed catalog [13], that proposes a paper-based set of pedagogical tasks/scenarios and associated tools.

A first experiment (a two months activity performed by two groups of 7 students) allowed us to validate the overall approach and the way we put it in practice. From a pedagogical point of view, the experiment demonstrated the very positive impact of the organizational features proposed by the framework on the students' activity, who used extensively the framework to work out their organization. As expected, we noticed a few occurrences of students not following exactly the *a priori* plan. In most cases they however explicited the new plan through the editor. *A posteriori* interviews allowed us to confirm the intermediation role that the plans and tasks description play.

The students did not encounter any difficulty with the plans and tasks notions (let us however recall that they are computer science students, this must be further analyzed with other publics). The general-purpose tools were used coherently with the tasks' descriptions (although students punctually used other means)[5], the software agents raised interesting issues (in particular, it appears that software agents must be fault-tolerant [15], much more than "real" students). The objective of this first experience was limited to studying if and how students used Symba, now that this is validated we plan to proceed to different experiments.

References

1. Bardram, J. E.: Plans as situated action: an activity theory approach to workflow systems. In: proceedings of ECSCW'97, Lancaster, UK (1997) 17-32
2. Bentley, R., Dourish, P.: Medium versus mechanism: Supporting collaboration through customisation. In: proceedings of ECSCW'95, Stockholm, Sweden (1995) 133-148
3. Betbeder, M-L., Tchounikine, P.: Une expérience d'activité collective médiatisée via le Web dans une FOAD, In: proceedings of TICE, Lyon, France (2002) 263-271
4. Bourguin, G., Derycke, A.: Integrating the CSCL activities into virtual campuses: Foundation of a new infrastructure for distributed collective activities. In: proceedings of Euro-CSCL, Maastricht, Netherlands (2001) 123-130
5. Dourish, P., Edwards, K.: A Tale of Two Toolkits: Relating Infrastructure and Use in Flexible CSCW Toolkits. Computer-Supported Cooperative Work, Vol.9(1) (2000) 33-51
6. Engeström, Y.: Learning by Expanding. An activity-theoretical approach to development research. Orienta-konsultit, Helsinki (1987)
7. Kuutti, K.: Activity Theory as a Potential Framework for Human-Computer Interaction Research. In Nardi B.A. (eds.), Context and Consciousness (1996) 17-44
8. Lewis, R.: Human Activities in Learning Societies.In: Proceedings of ICCE/ICCAI. Vol. 1, Tapei, Taiwan (2000) 36-45
9. Malone, T. W., Lai, K.-Y., Fry, C.: Experiments with Oval: A Radically Tailorable Tool for Cooperative Work. ACM Transactions on Information System, Vol. 13 (1994) 177-205
10. Morsh, A.: Three Levels of End-User Tailoring: Customization, Integration, and extansion. Third Decennial Aarhus Conference, Aarhus, Denmark (1995) 41-51
11. Morsh, A., Mehandjiev, N. D.: Tailoring as Collaboration: The Mediating Role of Multiple Representation and Application Units. In: proceedings of CSCW'2000, Kluwer Academic Publisher (2000) 75-100
12. Newell, A.: The Knowledge Level, Artificial Intelligence, Vol. 18 (1982) 87-127
13. Schneider, D., Dillenbourg, P., Frete, C., Morand, S., Synteta, P. : Le TecfaSEED Catalogue, manuel des enseignants (2002)
14. Suchman, L. A.: Plans and situated action: the problem of human-machine interaction. Cambridge University Press (1987)
15. Taurisson, N., Tchounikine, P.: Mixing Human and Software Agents: a Case Study. In: proceedings of ICALT, Athens, Greece (to appear) (2003)

[5] To what extent expliciting the organization and using the tools provided by the framework is presented as mandatory is a pedagogical issue, that concerns the human teachers that define and/or tutor the activity. For instance, in our 5th year of University context, the framework is a proposition: if the students are not satisfied with any aspect of the framework, they just skip to other means.