This article has been published as:

Tchounikine P. (2007). Directions to acknowledge learners' self-organization in CSCL macro-scripts. In: Groupware: Design, Implementation and Use, LNCS n°4715, Haake J.M., Ochoa S.F., Cechich A. (eds), Springer Berlin / Heidelberg.

Ref on Springer Website:
http://www.springerlink.com/content/v6208p17g3t61w34/

Directions to Acknowledge Learners' Self-Organization in CSCL Macro-Scripts

Pierre Tchounikine

LIUM, Université du Maine
Avenue Laënnec, 72085 Le Mans cedex 9, France
Pierre.Tchounikine@lium.univ-lemans.fr

Abstract. In this article we present a conceptual analysis of the notion of learners' self-organization in CSCL macro-scripts. We highlight that taking into account self-organization as an emergent feature of activity requires considering issues such as conceptual and technological tools to support learners' self-organization, maintenance of coherence between the script pedagogical objective and the emergent organization and between the technological setting and the emergent activity.

Keywords: CSCL; scripts; learners' self-organization; CSCL platforms.

1 Introduction

A Computer Supported Collaborative Learning (CSCL) *script* is an activity model designed to support a group of learners engaged in a computer-supported collective[1] task. A CSCL script defines the task to be achieved, the sequence of sub-tasks, learners' individual and collective roles, the technological setting and the constraints to be respected [1, 2]. Such scripts are defined to enhance the probability that knowledge-generative interactions such as conflict resolution, explanation or mutual regulation occur during the collaboration process [1, 3]. Technology platforms designed to support CSCL scripts typically provide communication tools (chat, mail, forum or whiteboard), awareness tools and task-related tools (tools related to the tasks to achieve) within static or workflow-oriented interfaces.

Scripts can be dissociated into macro-scripts and micro-scripts [2, 4]. A *macro-script* is a kind of pedagogical method. It aims at emphasizing the orchestration of activities by setting up a given set of conditions and constraints (e.g., sequence of individual and collective tasks learners must achieve or group characteristics) in order to trigger high-order thinking activities such as elaborating on content, explaining ideas and concepts, asking thought-provoking questions, constructing arguments, resolving conceptual discrepancies or cognitive modeling. A prototypical example is the Jigsaw-script family, i.e., scripts based on making individual learners manage some partial

[1] As we believe the "cooperative" / "collaborative" distinction is often a question of level of granularity, point of view and/or matter of concern, we will use "collective" as a wide concept, however using other authors' original wording when quoting specific works.

knowledge and then prompting them to solve collectively a problem that necessitates knowledge from each of them. Micro-scripts are finer-grained scripts studied at a psychological level, emphasizing the activities of individual learners and providing scaffolding such as sentence starters. Examples can be found in [1, 2].

Task (the prescribed work) and *activity* (what people actually do) are two different notions. Teachers set tasks, and learners interpret the specifications of the task, their subsequent activity being a more-or-less rational response to the task [5]. A script is a task-related notion. Learners' effective activity is related to different intertwined -and for some of them dynamic- issues. In this article we study one of these issues that may, in the context of macro-scripts in particular, play a role in learners' activity emergence: *learners' self-organization.*

2 Indirect-Design in Macro-Scripts

Considering CSCL scripts, 4 major dimensions can be pinpointed as artifacts that can be designed with the explicit objective of enhancing the probability that the targeted situations and interactions occur: (1) the didactic envelope [3], i.e., the set of pre-activities that allow triggering the script core mechanisms and contribute to create favorable conditions before interaction begins; (2) the script, i.e., the way groups, roles, tasks, resources, timing or constraints are defined; (3) the provided technological means; (4) the run-time regulations that aim at supporting collaboration by taking actions once the interaction has begun [6]: changing groups, relaxing constraints, modifying tasks, scaffolding, etc.

The didactic envelope, the script and the technological setting as designed by teachers set up the basic situation. The effective interaction pattern, i.e., the script as it unfolds as a set of activities and interactions taking place among the learners, is related to this basic situation (and then the regulation) but unpredictable in its details as different other dimensions from different natures may play a role: the general pedagogical and institutional context; the individual characteristics of learners; the effective motivation(s) of learners (e.g., play the game, please the teacher, solve the problem, interact with peers or gain social status) and the according effective activity/activities as related to their motivation(s); the social issues within the group such as the emergence of a leader or conflicting characters; etc. It is not possible *a priori* to exhaustively list and consider all the pedagogical parameters of a situation, the understanding of these parameters may vary (difficulty of a task, timing, balance of groups, etc.) and unpredicted events can happen (collapse of a group, pedagogical opportunity, external or technical constraint, etc.). Moreover, objects such as the script or the technological setting should not be thought of in terms of "objective" or "neutral" objects that all actors (learners, teachers) understand and consider in the same way. Learners' activity is related to the way they (as individuals, as a group) perceive and appropriate these issues for themselves, which may differ from teachers' or other students' perceptions. Technological characteristics in particular will be picked up in different ways by learners, which will appropriate them according to their purposes and in context [7]. And, things generally evolve and may vary run to run and/or during script enactment (understanding of the script, perception of the technological setting affordances, groups phenomena, motivation –e.g., using the proposed technological setting as a way to "play the game" and please the teacher but, when time runs out, focusing on completing the task satisfactorily in order to get a good mark and thus using other tools or means if more efficient).

Structuring activity is thus a challenging concept, in particular in the context of macro-scripts that leave some place for activity emergence. The script is a seed and a reference, but other dimensions play a role, and learners' activity can not be reduced and thought of just in terms of "playing the script": the way learners will perceive and enact the script can only be hypothesized. Emphasizing that designers have limited direct control over how their designs are enacted and that learning and learning environments thus can not be defined directly, [7] introduce the notion of *indirect design.* Within CSCL scripts, indirect design captures the idea that defining the script and the technological platform features and properties must be thought of as means to *influence* learners' activity, and this

activity (and the impact of the designed issues) must be taken into consideration as they happen and not as they were predicted by designers. This requires taking into account the different issues that may occur in such situations. Learners' self-organization dimension is one of these issues.

3 Macro-Scripts and Learners' Self-Organization

A collective phase of a script can be defined as a collective *work situation*, i.e., a situation where the learners are mutually dependent in their work. Works in CSCW highlight that actors engaged in such interdependent processes must address an overhead activity, that of articulating (dividing, allocating, coordinating, scheduling, meshing, interrelating) their respective activities [8, 9]. This is a meta-level overhead activity that is not focused on producing the targeted output, but on setting the conditions of the production of this output by maintaining a more-or-less stable pattern of cooperative arrangement between people. We will refer to this as learners' self-organization, "self" highlighting that, in our context, part of the organization is set by the script and part is related to learners' enactment of the script at run-time. We thus define learners' self-organization as the meta-level activity that a group of learners engaged in a CSCL script may engage in so as to maintain, within the reference frame that is externally defined by the script, a more-or-less stable pattern of collective arrangement.

Although they both relate to the learners' collective work structure, script and learners' self-organization differ in nature. A script is a prescriptive structure defined by the teacher. Learners' self-organization is an abstract inside-group feature that emerges from the way learners collaborate and enact the script, is influenced by —but different from— the script, and may vary run to run. A script mixes in an operational way what is to be achieved and how it is to be achieved: self-organization corresponds to what learners will make from this.

In CSCL scripts, the fact learners can or have to self-develop some organization is correlated to the script granularity and flexibility, i.e., the means and latitude that learners and teachers are proposed with in order to modify some script features such as groups, detailed subtasks decomposition, timing or technological setting [4]. The need and space for organization is almost non-existent in micro-scripts. Differently, if macro-scripts core-issues such as the general sequence of subtasks or their individual / collective nature are constrained by the script, different issues can be more-or-less left open to learners' self-organization, e.g., the precise timing, the name-by-name group composition and/or role attribution, some aspects of the tasks/subtasks decomposition (input and output, actors, roles, etc.) or the technology. As examples: in a jigsaw-script, learners can be allowed to compose the groups or modify them within the respect of the design decision stating that the group must be composed of individuals mastering different knowledge; within a setting such as grouping two learners and stating they have 3 hours to achieve a given subtask, different organizational possibilities are still open and learners can decide to spend 1 hour each on the same or on different issues and then share their thoughts, or explicitly split the 3 hours into different phases such as brainstorming, elicitation, argumentation and decision; learners may be allowed to choose the task-oriented tools, communication-tools or awareness-tools they want to use contextually, according to the emergent activity; etc. An analysis of the speech-acts detected in the mails, chats and forums used by learners while enacting a macro-script run for 6-7 weeks at University level highlighted that 20% to 50% (3 groups analyzed) were related to organizational issues such as organization proposals (e.g., "The first issue is to define the objective of this phase", "Would you agree to work on the basis of the graphic?" or "Each could be put in charge of …"), refinements, agreements or disagreements linked to these proposals, and of course organizing meetings or managing time [10]. Although this is a border-line situation (very coarse-grained and long-period script with University students) whose results are not to be generalized, it emphasizes the potential importance of learners' self-organization in script enactment.

4 Directions to Acknowledge the Self-Organization Notion

4.1 Considering the Setting Potential Self-Organization Issues

Focusing on organization, macro-scripts carry out a tension between different issues from which: (1) a script carries constraints that define boundaries for, and impact on, learners' eventual self-organization; (2) a script is defined by teachers; (3) a script should be easily appropriable by learners [3] (macro-scripts suppose a high commitment of learners: they create a didactical contract between the teachers and the learners and between the learners, and there is an assumption that learners will "play the game"); (4) the fact that people appropriate a structure for themselves and/or develop a shared understanding of it is generally related to how much the structure has been collectively constructed and/or refined; (5) organization is a structure that emerges, is instable and evolves during activity; (6) the technological platform may impact (constrain, allow, support) learners' activity and self-organization.

Learners' self-organization and flexibility are related issues: the flexibility that is left to learners impacts how self-organization can emerge, and under what constraints. Given the setting, taking self-organization into account can appear necessary to avoid counterproductive issues and/or be used as a means to contribute with the script in making the targeted knowledge-generative interactions appear. Leaving open to learners some organizational issues may present some interest related to learners' appropriation of the script by becoming active actors in organizing work and/or tackling unanticipated problems such as a time management problem, a learner that abandons or downloads his contribution, a bad role distribution or an inadequate technological decision. It may also be an objective *per se*, for instance as a means to make learners practice and learn how to work collectively.

Considering learners' self-organization raises questions such as: what relations can be drawn between self-organization issues and the learning targeted by the script, and/or some other high-level skills such as autonomy or collective work skills? How can one perceive and then deal with a learners' self-organization that diverges from the script pedagogical objectives? How to understand and deal with the dynamics of learners' self-organization and the central notion of breakdown [8] which can be both an opportunity for learning and/or the cause of a collapse of the learning situation? How must one deal with the fact that self-organization is an activity in itself, that can be intertwined with others but may also interfere with the flow of work? How can learners' self-organization be impacted (influenced, supported)?

As learners' self-organization is fundamentally an emergent feature, it is not to be addressed in terms of prediction and direct design but in terms of indirect design and regulation. As activity in general (cf. *supra*), self-organization is related to different complex intertwined factors which renders it difficult to predict. Some of these factors can however be impacted: the didactic envelope, the script structure and/or the technological characteristics may be used as direct or indirect means to provide seeds, opportunities and incentives. And, teachers in charge of running the process should have means to perceive the learners' activity and act if it appears that the emergent organization goes against the script pedagogical objectives.

4.2 Tools to Support Learners' Self-Organization

A group facing a self-organization situation undertakes an activity (in the sense of Activity Theory [11]) that is linked to a motivation (establish a pattern of cooperative arrangement). Interestingly, it is a collective activity. As an activity, it is mediated by tools from which we can distinguish conceptual tools and technological tools.

A basic CSCL script setting implicitly carries potential conceptual tools that learners can use to conceptualize the collective work of organizing themselves and reflecting on the setting: the epistemic notions used by teachers to describe the script (e.g., "group" or "role"). The fact that self-organization is usually not considered as a specific concern is widely related to the implicit basic

assumption that scripts structure learners' activity and, if any adjustment is required, learners can complement the structuring using the same conceptual notions and the available communication and awareness functionalities. This is however to be questioned: there is an issue in understanding what conceptual and technological tools can support self-organization and how, and if/how this support can be correlated with the script in order to influence activity in a way that is coherent with the pedagogical objectives. As an example, [10] reports a work where the script is presented to learners using coarse-grained notions (phases, tasks) but the technological platform provides editors allowing learners to render explicit and discuss how they intend to tackle these tasks using notions inspired from Engeström's triangle [11] (subtasks defined in terms of a subject, an object, a community and rules, tools, and division of labor), which appears to limit the risk for the script to collapse by lack of organization and commitment.

4.3 Maintaining a Coherence Script-Objectives / Emergent-Organization

Learners' self-organization and more generally learners' activity can be analyzed with respect to the script, but more interestingly with respect to the *script pedagogical objectives* as the way these objectives have been reified in the actual setting may be linked to contingent issues. As discussed previously, scripts carry a tension between structuring learners' processes and supporting knowledge-generative interactions. Learners' self-organization (among other issues) may conduct them to diverge from the teacher's *a priori* script: is this a problem?

In order to tackle this issue and allow flexibility and learners' self-organization whilst not contradicting pedagogical objectives, the conceptual dissociation between scripts *intrinsic* and *extrinsic constraints* proposed in [4] appears useful. Intrinsic constraints denote the script core mechanisms, e.g., within a jigsaw script learners must manage different knowledge. Extrinsic constraints are contingent decisions related to detailed groups composition, roles attribution, tasks/subtasks definition or technological setting. Extrinsic constraints define the space for flexibility, i.e., the space within which the structure carried by a script should be modifiable by learners and/or teachers because the related decisions result from arbitrary or practical choices. Intrinsic constraints set up the limits of flexibility, i.e., what cannot be accepted in order for the script to keep its *raison d'être* [4]. This dissociation provides a substratum to address automated and/or teacher regulation issues such as analyzing learners' self-organization with respect to the script objectives (as opposed to contingent issues), allowing and eventually supporting learners' actions that diverge from the script but remain coherent with the pedagogical objectives, acting on the setting (adapting the script, adapting the platform) or dealing with situations that go against some pedagogical objectives. The technological platform should be capable of permanently comparing the difference between the script and the actual interaction pattern (and/or helping the teacher to do so, and/or the learners) in order to check if eventual differences violate some intrinsic constraints. This requires the platform to maintain (1) a model of the script and underlying *design rationale* and (2) a model of the script enactment, updated in real time, these 2 models being interoperable [4].

4.4 Maintaining a Coherence Technological-Setting / Emergent-Activity

CSCL scripts can be associated with a generic Learning Management System (LMS) platform, i.e., a kind of generic technological platform that proposes some general-purpose tools (e.g., chat, email, shared agenda or file exchange zone). Associating scripts with platforms that are studied (designed, customized) according to the script, however, presents a list of advantages, in particular (1) the platform can be process-oriented and reify (part of) the script features (sequence of tasks, constrained access to resources, roles, etc.) and (2) the platform can propose tools that are specifically adapted to the context, the tasks to be achieved by learners and/or pedagogical objectives. Technological settings can thus be studied as a means for the two dual dimensions of a script, supporting and constraining learners' activities.

When targeting a script-related platform, the design should consider the script and the learners' activity, which may vary (in particular because of self-organization) from the script-related expectations. However, designing a platform to support activity is somewhat paradoxical as activity will emerge and is not fully predictable. An approach to this issue is to target tailorable platforms. A computer system is said to be tailorable if it proposes its users with some means to modify itself in the context of its use, as one of its functionalities [12]. In CSCL scripts, tailorability is a means to allow learners to adapt the platform to needs in context, according to how the script is enacted and the underlying emergent issues such as organizational issues if any. Introducing tailorability features in CSCL platforms however raises three major issues: (1) tailorability for learners is to be studied with respect to the scope of flexibility defined by the intrinsic/extrinsic constraints notions, and teachers' regulation [4]; (2) tailorability must be technically easy; (3) tailorability is, with respect to the learners' activity as related to the script, *another* activity; there is therefore a risk of causing a breakdown in the activity flow.

5 Conclusions

The results presented in this article are a definition of the notion of learners' self-organization, an argumentation that it is an intrinsic dimension of CSCL macro-scripts enactment that, if not acknowledged, may conduct not to reach the script pedagogical objectives, and directions to acknowledge this notion.

In section 4 we have raised different important issues to acknowledge learners' self-organization. All of these imply a detailed machine-readable modeling of the script. Different general [13] or CSCL-focused [14, 15, 16, 17] Learning Design languages allowing to model scripts have recently been proposed. From the perspective of learners' self-organization, such languages are to be considered with respect to the fact (1) they allow to model script notions that are in relation with self-organization and (2) the implementation approach allows the required flexibility. From the point of view of representation, CSCL-focused languages introduce scripting concepts (groups, roles, etc.) which facilitate making explicit organization issues (in general), and thus may facilitate learners' self-organization identification and/or representation (in settings that consider this objective). From the point of view of semantics, the issue is that of ensuring the relation between the script, its enactment and the technological platform, and the means provided to learners for flexibility. Mechanisms such as generating the platform from a graphical/formal description of the script as proposed in LAMS [17] or in [15] can allow to provide interesting run-time flexibility means for the teacher and, if extended to them, for the learners. Within such an approach, the issues are (1) the granularity of the modeling language as going into details of scripts modeling of course raises the computer-science difficulty of insuring the script/platform coherence, (2) the conditions of access to the editing (who, when, how), (3) the usability of the editor for end-users and (4) the management of the underlying coherence issues. Some of the state-of-the-art techniques thus do allow partially tackling some the issues we have raised, however not in a straightforward, articulated nor complete way.

This work has benefited from fruitful discussions within the CSCL groups of Kaleidoscope, a European Network of Excellence for Technology Enhanced Learning.

6 References

1. Fischer, F., Mandl, H., Haake, J., Kollar, I.: Scripting Computer-Supported Collaborative Learning – Cognitive, Computational, and Educational Perspectives. Computer-Supported Collaborative Learning Series. Springer (2007)
2. Kobe, L., Weinberger, A., Dillenbourg, P., Harrer, A., Hämäläinen, R, Fischer, F.: Specifying collaboration scripts. IJCSCL (to appear)
3. Dillenbourg, P., Jermann, P.: Designing integrative scripts. In: Fischer, F., Mandl, H., Haake, J., Kollar, I. (eds) Scripting Computer-Supported Collaborative Learning – Cognitive, Computational, and Educational Perspectives, pp 275--301. Springer (2007)

4. Dillenbourg, P., Tchounikine, P.: Flexibility in macro-scripts for CSCL. In: Journal of Computer Assisted Learning 23(1), 1--13 (2007)
5. Goodyear, P.: Effective networked learning in higher education: notes and guidelines. http://csalt.lancs.ac.uk/jisc/
6. Soller, A., Martinez, A., Jermann, P., Muehlenbrock, M.: From Mirroring to Guiding: A Review of State of the Art Technology for Supporting Collaborative Learning. IJAIED 15, 261--290 (2005)
7. Jones, C., Dirckinck-Holmfeld, L., Lindström, B.: A relational, indirect, meso-level approach to CSCL design in the next decade. IJCSCL 1 (1), 35--56 (2006)
8. Bardram, J.: Designing for the Dynamics of Cooperative Work Activities. In: Poltrock, S., Grudin, J. (eds) CSCW conference, pp 89--98. Seattle (1998)
9. Schmidt, K., Bannon, L.: Taking CSCW Seriously: Supporting Articulation Work. CSCW 1(1-2), 7--40 (1992)
10. Betbeder, M-L., Tchounikine, P.: Symba: a tailorable framework to support collective activities in a learning context. In: Favela J., Decouchant, D. (eds) CRIWG 2003. LNCS, Vol. 2806, pp 90--98. Springer (2003)
11. Engeström, Y.: Learning by expanding. An activity-theoretical approach to developmental research. Helsinki: Orienta-Konsultit (1987)
12. Morsh, A.: Three Levels of End-user Tailoring: Customization, Integration, and Extension. In: Kyng, M., Mathiassen L. (eds) Computers and Design in Context, pp 51--76. The MIT Press, Cambridge MA (1997)
13. IMS-LD, http://www.imsglobal.org/learningdesign
14. Ferraris, C., Martel, C., Vignollet, L.: LDL for Collaborative Activities. In: Botturi, L., Stubbs, T. (eds) Handbook of Visual Languages in Instructional Design: Theories and Practices. Hershey, PA: Idea Group (in press)
15. Haake, J., Pfister, H.-R.: Flexible scripting in net-based learning groups. In Fischer, F., Kollar, I., Mandl, H., Haake, J.M. (eds.) Scripting computer-supported cooperative learning – Cognitive, computational, and educational perspectives. Springer (2007)
16. Harrer, A., Malzahn, N.: Bridging the gap – towards a graphical modeling language for learning designs and collaboration scripts of various granularities. In: Kinshuk (ed) ICALT'2006, pp 296--300. Kerkrade (2006)
17. LAMS, http://www.lamsinternational.com