



HAL
open science

Specifying computer-supported collaboration scripts

Lars Kobbe, Armin Weinberger, Pierre Dillenbourg, Andreas Harrer, Raija Hämäläinen, Frank Fischer

► **To cite this version:**

Lars Kobbe, Armin Weinberger, Pierre Dillenbourg, Andreas Harrer, Raija Hämäläinen, et al.. Specifying computer-supported collaboration scripts. *International Journal of Computer-Supported Collaborative Learning.*, 2007, 2(2-3), pp.211-224. hal-00703944

HAL Id: hal-00703944

<https://hal.science/hal-00703944>

Submitted on 4 Jun 2012

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

SPECIFYING COMPUTER-SUPPORTED COLLABORATION SCRIPTS

This is a post -print of an article submitted for consideration in the International Journal of Computer-Supported Collaborative Learning © 2007 Springer Verlag.

Personal use of this manuscript is permitted. Permission from Springer Verlag must be obtained for any other commercial purpose.

This article may not exactly replicate the published version, due to editorial changes and/or formatting and corrections during the final stage of publication. Interested readers are advised to consult the published version which can be found at:

<http://www.springerlink.com/content/6286w8235q818664/>

doi: { 10.1007/s11412-007-9014-4 }

Please refer this manuscript as:

Kobbe, L., Weinberger, A., Dillenbourg, P., Harrer, A., Hämäläinen, R., Häkkinen, P., et al. (2007). Specifying computer-supported collaboration scripts. *International Journal of Computer-Supported Collaborative Learning*, 2(2-3), 211-224.

Running Head: SPECIFYING COLLABORATION SCRIPTS

Specifying Computer-Supported Collaboration Scripts

Lars Kobbe and Armin Weinberger

Knowledge Media Research Center, Tübingen, Germany

Pierre Dillenbourg

Swiss Federal Institute of Technology, Lausanne, Switzerland

Andreas Harrer

University of Duisburg-Essen, Germany

Raija Hämäläinen

University of Jyväskylä, Finland

Frank Fischer

Knowledge Media Research Center, Tübingen, Germany

Author Note

Lars Kobbe, Armin Weinberger, Ph.D., & Prof. Frank Fischer, Research Unit on Collaborative Knowledge Building; Andreas Harrer, Ph.D., Institute for Computer Science and Interactive Systems; Prof. Pierre Dillenbourg, CRAFT – Center for Research and Support of Training and its Technologies; Raija Hämäläinen, Research Group on Collaboration and Authenticity in Open Technologically Enriched Learning Contexts.

This research was funded by the European Network of Excellence “Kaleidoscope”.

Point of contact: Lars Kobbe, Konrad-Adenauer-Str. 40, 72072 Tübingen (Germany),
Phone:++49 (0)7071 979-202, Fax:++49 (0)7071 979-200, email: l.kobbe@iwm-kmrc.de

Abstract

Collaboration scripts are activity programs which aim to foster collaborative learning by structuring interaction between learners. Computer-supported collaboration scripts generally suffer from the problem of being restrained to a specific learning platform and learning context. A standardization of collaboration scripts first requires a specification of collaboration scripts that integrates multiple perspectives from computer science, education and psychology. So far, only few and limited attempts at such specifications have been made. This paper aims to consolidate and expand these approaches in light of recent findings and to propose a generic framework for the specification of collaboration scripts. The framework enables a description of collaboration scripts using a small number of components (participants, activities, roles, resources and groups) and mechanisms (task distribution, group formation and sequencing).

Specifying Computer-Supported Collaboration Scripts

Successful collaborative learning depends upon effective interaction amongst learners. However, when learners are left to their own devices, they rarely engage in productive interactions such as asking each other questions, explaining and justifying their opinions, articulating their reasoning, or elaborating and reflecting upon their knowledge. Collaboration scripts aim to foster collaborative learning in shaping the way in which learners interact with one another. In specifying a sequence of learning activities, together with appropriate roles for the learners, collaboration scripts are designed to trigger engagement in social and cognitive activities that would otherwise occur rarely or not at all.

Collaboration scripts are based upon the *scripted cooperation* approach, as described by O'Donnell (1999), which differs from other collaborative learning approaches chiefly in the fact that it focuses on the specific activities that learners are expected to engage in, whereas others leave them unspecified or vague. In targeting those activities which have emerged from research findings in cognitive and educational psychology as being strongly related to learning (e.g., Cohen, 1994; Webb & Palincsar, 1996), scripts are assumed to lead to higher level cognitive processing and therefore to better learning outcomes.

Collaboration scripts have become fairly popular within educational science, especially in the domain of computer-supported collaborative learning (CSCL), where they have been used in various settings, including face-to-face, web-based as well as mobile contexts (MOSIL, 2004). Computer-supported collaboration scripts carry the additional benefit of reducing the coordinative effort both on the teachers' and the students' part. Computers may for instance keep track of the students' position within the script sequence, alert or prompt students to engage in specific activities and provide additional information and resources when needed. Furthermore, once programmed, computer-supported scripts can be reused as many times and with as many students as desired. However, whereas traditional collaboration scripts can easily be customized and adapted to new contexts, computer-

supported scripts generally suffer from the problem of being restrained to a specific learning platform and learning context. Scientists with an interest in educational modeling languages are currently aiming to formalize collaboration scripts in order to foster reusability and portability of standardized scripts. However, there first needs to be a clear understanding of how collaboration scripts work and what they are composed of. Specifying collaboration scripts requires the integration of multiple perspectives from computer science, education as well as psychology. So far, only very few and limited attempts at specifying collaboration scripts have been made. The goal of the present paper is to consolidate and expand these approaches in light of recent findings and to propose a generic framework for the specification of collaboration scripts.

Approaches to the Specification of Collaboration Scripts

In a pioneering attempt to analyze collaboration scripts, Dillenbourg (2002) identified a number of aspects which served as a preliminary framework for script comparison and design. Dillenbourg (2002) described scripts as a sequence of phases, each characterized by the following five attributes: *type of task* to be accomplished, *group formation* (and composition), *distribution of the task* within and among groups, *type and mode of interaction* (e.g., co-located vs. remote, synchronous vs. asynchronous, text-based vs. voice-based, etc.), and *timing* of the phase. From phase to phase, each of these attributes can change. The allocation and re-allocation of roles and activities, as well as physical or virtual resources are considered to be part of the task distribution.

In their conceptual framework, Dillenbourg and Jermann (in press) expanded the scope of collaboration scripts to encompass more than just small group interaction. Whilst collaborative activities are still regarded as being essential to the learning process and constituting the core learning mechanism, *integrative scripts* also include individual activities and activities encompassing a whole class. Thus, collaborative *core scripts* are positioned within a *didactic envelope*, that is, pre- and post-structuring activities which enable scripts to

be optimally integrated into the lesson plan (e.g., introducing the topic, reflecting on what was discussed, etc.) and which contribute to their effectiveness and consistency. The integrative aspect of scripts becomes even more evident in the case of computer-supported collaboration scripts, in which scripts integrate virtual with physical activities and manage the data flow between them. For example, in the ArgueGraph Script (cf. appendix C), discussion across a whole class is based on a graph that was computed from individual responses priorly collected in the form of a questionnaire.

According to Dillenbourg and Jermann (in press), the didactic envelope features two salient dimensions: The *time structure* and the *social structure* of scripts. Each activity within a script can be mapped onto a certain phase or point of time in the overall sequence and is also limited in its duration (giving the time structure). Furthermore, each activity can be mapped onto a specific *social plane* (in reference to Vygotsky, 1978), such as an individual, group, class, community or world plane (giving the social structure). The didactic envelope manages the integration of activities from phase to phase and from plane to plane.

The *core script* typically involves a task distribution among all group members. Dillenbourg and Jermann (in press) viewed interaction as the means of overcoming these task divisions and proposed a model, according to which the nature of the task distribution also determines the nature of the members' interaction (e.g., explanation, argumentation, mutual regulation, etc.). Following Schwartz' (1995) definition of collaborative learning as the effort necessary to build a shared understanding, learning results from the interactions which students engage in to build a shared understanding despite the fact that the task is distributed (Dillenbourg & Jermann, in press). According to the design principle "Split Where Interaction Should Happen", the model was named "SWISH".

Kollar, Fischer and Hesse (in press) were interested in comparing scripts from two research traditions: Computer-supported collaborative learning (CSCL) and instructional psychology. Five conceptual components were identified according to which scripts could be

differentiated: *Objectives, activities, sequencing, roles, and types of representation*. Scripts from both research traditions had similar objectives regarding the group task. However, as all of the CSCL scripts made use of computer-mediated communication, a major part of these scripts was devoted to the support of smooth and coherent communication and coordination with respect to the inherent weaknesses of the medium (e.g., Herring, 1999). Although scripts did not differ much in the kind of activities which they promoted (e.g., arguing, explaining, question asking), scripts from instructional psychology gave much more support when it came to how to carry out these activities (through training or detailed instruction). CSCL scripts typically lacked a prescribed sequential structuring of activities, although the interface sometimes implicitly induced a certain sequence of activities. Also, CSCL scripts often lacked clear role distribution in contrast to scripts from instructional psychology. Scripts from both traditions were rather inconsistent in the way which they were represented during the learning phase (e.g., represented internally in the learners' mind or represented externally by buttons on the screen, prompt cards, etc.). These findings highlight the diversity of scripts, both in realization as well as in function and purpose.

Within a conceptual framework, Kollar et al. (in press) used a distributed cognition approach in order to incorporate the five components mentioned above: because the group task is solved or the *objective* is reached by engaging in certain activities, it can also be defined by a specific set of *activities*. In a script, *activities* can also be distributed among the learners in the form of *roles*. These activities and roles are linked by a specific *sequence* following a meaningful strategy and ultimately leading to the solution of the task. The script is to a certain degree *represented* externally (e.g., written, graphical or oral) and internally (as procedural knowledge).

The Role of Activities in Collaboration Scripts

Collaboration scripts essentially concern activities that promote learning, but which rarely occur spontaneously within the discourse of learners (O'Donnell, 1999). In this way,

collaboration scripts aim to directly facilitate specific activities during collaborative learning rather than setting the conditions for collaborative learning prior to a collaborative learning phase (e.g., establishing a specific incentive structure; Slavin, 1983). Within collaborative discourse different kinds of activities have been identified, that are particularly beneficial to learning (cf. King, 1999; Ohlsson, 1995; Weinberger & Fischer, 2006). Some of the activities most recurrently referred to are elaboration, explanation, argumentation and question asking.

Elaboration is characterized by relating new ideas and concepts to that which is already known, making it personally more meaningful, and expanding it in multiple ways, such as adding details, giving examples, making analogies, creating visualizations and predicting outcomes. The benefits of elaboration to learning are well documented (e.g., van Boxtel, van der Linden, & Kanselaar, 2000). High quality elaboration involves a reorganization of the learner's knowledge and can for instance be promoted by contrasting concepts, rather than by simply creating personal examples or expanding on the effects of concepts (Hamilton, 1997).

Good explanations go beyond simply giving a step-by-step solution in so far as they also articulate the explainer's reasoning and elaborate on the concept (Bargh & Schul, 1980). Giving explanations fosters learning in that it prompts learners to check for inconsistencies and knowledge gaps as well as to reorganize and clarify the material in order to meet the target audience's knowledge level (e.g., Webb, 1989). In order to benefit from explanations which they receive, learners should try to apply the help given until they reach the point where they are able to solve a problem by themselves (Webb, Troper, & Fall, 1995).

Argumentation involves the generation of claims or assertions and their justification with evidence. Although arguments are primarily intended to convince someone else of one's own position, they are also effective in promoting self-explanations and deeper understanding of a subject (Baker, 2003). In constructing arguments, learners must actively search for knowledge with which they can support their claim. In order to arrive at strong arguments

which cannot easily be refuted, learners need to reflect upon their own understanding and check for possible ambiguity and inconsistencies in their statements. Furthermore, collaborative learners construct and exchange arguments, counter-arguments and replies in discourse, in which learners challenge and refine their knowledge (Leitão, 2000).

Question asking can foster comprehension by elaborating on content and can furthermore help to monitor comprehension by checking for errors in understanding (Palincsar & Brown, 1994; King, 1994). It can also be used for eliciting information that is needed in order to fill in gaps in knowledge (as in “help seeking”). Question asking is frequently assigned as an activity in collaborative learning. However, when learners are simply asked to engage in question asking, they typically ask rather simple, factual questions that require little cognitive effort. Research has identified a number of strategies (e.g., providing generic example questions or making use of question starters) to help learners generate more effective, thought-provoking questions (cf. Rosenshine, Meister, & Chapman, 1996, for an overview).

To conclude, the role of activities in collaborative learning is of central importance, since the type of activities induced has a strong impact on the cognitive processes evoked. Nevertheless, it is not simply what the learners do but also how they do it, that is predictive of the learning outcomes. While all collaboration scripts aim at promoting these desired activities in the learners’ interaction, there appear to be general differences that lead researchers to distinguish macro- from micro-scripts (e.g., Dillenbourg & Jermann, in press). Macro-scripts typically differ from micro-scripts in the level of granularity (i.e., activities typically describe longer time segments and are spread over more social planes compared to micro-scripts). Micro-scripts, on the other hand, tend to provide more scaffolding to students such as sentence starters, question prompts or descriptions (cf. Kollar et al., in press). Generally speaking, macro-scripts take a rather top-down and pedagogical approach to structuring interaction (emphasizing the orchestration of activities within the classroom),

whereas micro-scripts take a more bottom-up and psychological approach (emphasizing the activities of individual learners). For all apparent differences, both macro- and micro-scripts seem to share the same compositional structure and can therefore be described with the same set of components and mechanisms.

Specifying Collaboration Scripts – A Framework

Based on the frameworks presented by Dillenbourg (2002), Dillenbourg and Jermann (in press) and Kollar et al. (in press), we aim to propose a framework that consolidates recent approaches with current ideas and conceptualizations of our research.

Our framework distinguishes between different levels of abstraction for the specification of scripts. At the highest level of abstraction are the so called script schemata (Dillenbourg & Jermann, in press). Schemata describe the core design principle through which the script is expected to trigger specific interactions. For instance, many scripts use a variation of the Jigsaw method (Aronson, Blaney, Stephan, Sikes, & Snapp, 1978) and form pairs with complementary knowledge, providing them with complementary information or roles in order to create socio-cognitive conflict, or assign and alternate roles which foster reciprocal activities such as questioning or tutoring. Thus, the core design principles are referred to as the Jigsaw schema, the conflict schema and the reciprocal schema (Dillenbourg & Jermann, in press). Besides these aforementioned schemata, it is expected that further schemata exist, which we aim to identify in the future. However, for the purpose of specification and subsequent formalization, we target a level of abstraction that is below that of script schemata and is still generalizable enough to allow for adaption to the learning context, for example, the number of participants and the kind and content of resources. Dillenbourg and Jermann (in press) refer to this level as the level of script classes, which cover a range of scripts representing acceptable variations of a prototype. For example, the Universanté Script (cf. appendix B) is a prototype of the Jigsaw schema, whereas the ArgueGraph Script (cf. appendix C) is a prototype of the conflict schema.

Regarding the structural composition of scripts, we distinguish between mechanisms on the one hand and components on the other. This distinction has not been made in earlier approaches. For example, Kollar et al. (in press) regarded *role distribution* as a script component, whereas we refer to roles as a component and distribution as a mechanism. Furthermore, we do not regard earlier components such as *type of representation* or *mode of interaction* as either mechanism or component, because these aspects are merely design decisions that can be used to build variations of one and the same script class. The script components and mechanisms will be presented separately in the following sections and will be illustrated with script examples in the appendices.

Script Components

Our framework aims to be economic in enabling scripts to be described with just a small number of components: the individuals that participate in a script, the activities that they engage in, the roles they assume, the resources that they make use of and the groups they form.

Participants. The term participants is used synonymously with users, persons or people, in other words, as a general abstraction of concrete individuals. Scripts usually have certain requirements regarding the total number of participants that they can handle, sometimes given as a variable range (e.g., from three to eight participants) or as a multiple of another script component (e.g., two participants per text book). Scripts may also take into account specific participant characteristics, such as individual opinion or knowledge in a domain. For example, the ArgueGraph Script (cf. appendix C) makes use of participants with different opinions, whereas the Universanté Script (cf. appendix B) utilizes participants with different background knowledge.

Activities. Activities form a hierarchical structure in which any greater activity can be decomposed into lesser, more fine-grained activities, and any lesser activity can be subsumed by one or more greater, more coarse-grained activities. For instance, discussing can be

decomposed into explaining, constructing arguments, question asking, etc., and asking somebody to check a report for mistakes can be generalized as help-seeking. As discussed before, the kinds of activities specified are highly relevant to the type and degree of learning taking place (cf. King, in press).

Roles. A simple function of roles in collaboration scripts is to refer to specific participants when assigning activities or allocating resources. However, roles also provide a kind of legitimation and are associated with privileges, obligations and expectancies. For example, a person in the role of the critic has the legitimation to criticize another student's work, the privilege to speak out frankly, the obligation to justify the critique and is expected to point out shortcomings as well as ways of improvement. As roles are closely tied to learning activities, their title is usually predictive of the activities which participants are allowed, obliged or expected to engage in. Thus, specific roles may foster particular learning activities, such as a "scientist" promoting planning, observing and drawing conclusions. Participants may assume one or more roles at any given time and can exchange these roles with other participants. Even if no explicit role assignment takes place, collaboration scripts may implicitly assume a specific "non-role condition" in order to prevent counterproductive self-appointment of roles, for instance, it is often not desirable for small groups to allow one participant assume the self-appointed role of group leader.

Resources. Resources in general comprise virtual or physical objects which can be allocated to learners (e.g., text books, tools, etc.). Some resources are stable, predefined objects whereas others are created or significantly modified during the script, such as filled-out questionnaires and written notes. Within this framework, resources are not restricted to tangible objects, but may in effect encompass everything the learning activities operate on, such as verbal and non-verbal productions. Although resources have not been identified as script components in earlier approaches, they are important within the context of scripts because they often constitute a common object of or reason for interaction. Resources often

foster interaction in that they are distributed amongst learners in a way that induces social interdependence (i.e., each learner needs access to the others' resources), a method commonly employed in collaborative learning scenarios such as Jigsaw (Aronson et al., 1978) and conceptualized in the SWISH-model by Dillenbourg and Jermann (in press).

Groups. Groups generally form a hierarchical structure with larger groups being composed of one or more smaller groups. Participants can be grouped in terms of existing common features (e.g., gender, age or national groups) or can be distributed into new groups based on certain criteria such as desired group size, amount or composition (see the group formation mechanism). Participants can also become members of multiple and overlapping groups.

Script Mechanisms

Script mechanisms help to describe the distributed nature of scripts, that is, how activities, roles and resources are distributed across participants (*task distribution*), how participants are distributed across groups (*group formation*) and how both components and groups are distributed over time (*sequencing*). Each of these mechanisms features particular principles which are important for issues of scalability (e.g., for applying the same script to a varying number of participants) and later formalization.

Task Distribution. A key feature of scripts is the distribution of the task, such as providing participants with only one piece of the information that they need in order to foster knowledge exchange with one another. Decomposable activities can also be distributed in such a way that one learner engages in a cognitive activity while the other learner engages in a supportive metacognitive activity. Distribution typically makes use of a function, in other words, the elements of one set (e.g., roles) are mapped through a specific function (e.g., one-to-one) onto the elements of another set (e.g., participants). In some cases, this function may become quite complex, such as arranging roles, participants and groups in such a way that

each participant assumes a different role in each group (cf. the Universanté Script in appendix B).

Group Formation. While some groups just happen to exist by definition (e.g., gender groups such as men and women or hair color groups such as blondes, brunettes and redheads), other groups must be formed by a particular procedure or principle. In most cases, group formation is very simple, such as forming groups by amount (e.g., dividing a class into four groups), by size (e.g., dividing a class into groups of four) or by a combination thereof (e.g., forming four groups of four). Some scripts make use of more complicated principles that do not only take into account group numbers and size, but also the composition of each group and the overall balance among the groups. The Universanté script, for instance, forms groups based on the principle of obtaining at least one group for each case description with at least two participants from each nation and all participants being balanced evenly across the case groups (cf. appendix B).

Sequencing. Collaboration scripts provide a temporal structure for interaction, that is, they specify the order in which events and activities are to take place. Script sequences are sometimes divided into distinct segments, referred to as *phases*, that provide a helpful point of reference for students and teachers and are frequently used for managing time. Scripts are not limited to a strictly linear sequence of activities, but may also feature complex sequential arrangements with loops and branches. Most frequently, collaboration scripts make use of repetitions with minor variations: In the case of the MURDER Script (cf. appendix D), for instance, the text passage and role allocations change with each cycle. Further common sequencing patterns are traversal, rotation and fading. Traversal describes a sequence in which all elements of one set are looped through, with only one element being in use at any given time (the text passage in the MURDER Script). This is often used to give students the chance to practice the same activities on different sets of data. Rotation permutes the order of elements in a given set (the roles in the MURDER Script) and is frequently used to give

each student the opportunity to engage in each activity. Fading refers to features that are gradually added (faded in) or removed (faded out) from a script (in the MURDER Script, setting the mood may just as well be omitted after the initial cycle) and is often used to gradually increase or decrease the degree of instructional support.

Conclusions

Beginning with a review of the original conception of scripts for collaborative learning, we presented recent approaches towards a specification of collaboration scripts. Since the instructional approach of scripting interaction is essentially a cognitive approach, we discussed the kind of activities which foster learning from a cognitive point of view. Besides activities, we also presented a number of other essential components and mechanisms that constitute a script.

The proposed framework serves a variety of different purposes. First of all, it provides a common terminology as the cornerstone for knowledge exchange and accumulation in scientific research on scripting collaborative learning. It is aimed at facilitating the systematic exploration of script mechanisms as well as the comparison and integration of research results. Nevertheless, the framework does not impose a specific theoretical perspective on the interplay of script components and mechanisms but is rather intended to be useful within a wide range of theories. For practitioners, the components and mechanisms elaborated here may serve as a checklist for the design of scripted CSCL environments, as they are of high relevance to collaborative learning. We also expect the prototypical examples to be useful as models of good practice.

In the near future, we aim to go beyond the semi-formal description demonstrated in the script examples to a formal modelling language that can be used in computer-supported learning environments. Because the linear structure of text-only representations is ill-suited to model concepts such as rotation, fading or traversal, at least part of the formal language will use a graphical notation. The use of statechart diagrams for the modelling of collaboration

scripts is currently being explored and has so far shown promising results. Nevertheless, we believe that a semi-formal script description is more than just a transitional step in the formalization process and is in itself valuable: Many researchers and practitioners (even in the field of technology enhanced learning) may not feel at ease with reading formal, computational languages, let alone applying this formalism in transcribing scripts which they have designed themselves. Thus, a semi-formal description may serve as an intermediate representation that bridges the gap between differing demands.

References

- Aronson, E., Blaney, N., Stephan, C., Sikes, J., & Snapp, M. (1978). *The jigsaw classroom*. Sage Publications.
- Baker, M. (2003). Computer-mediated argumentative interactions for the co-elaboration of scientific notions. In J. Andriessen, M. Baker & D. Suthers (Eds.), *Arguing to learn: confronting cognitions in computer-supported collaborative learning environments* (Vol. 1, pp. 1-25). Dordrecht: Kluwer.
- Bandura, A. (1997). *Social learning theory*. Englewood Cliffs, NJ: Prentice-Hall.
- Bargh, J. A., & Schul, Y. (1980). On the cognitive benefits of teaching. *Journal of Educational Psychology, 72*, 593-604.
- Cohen, E. G. (1994). Restructuring the classroom: Conditions for productive small groups. *Review of Educational Research, 64*(1), 1-35.
- Chi, M. T. H., deLeeuw, N., Chiu, M., & LaVanher, C. (1994). Eliciting self-explanations improves understanding. *Cognitive Science, 18*, 439-477.
- Dansereau, D. F., Collins, K. W., McDonald, B. A., Holley, C. D., Garland, J. C., Diekhoff, G., & Evans, S. H. (1979). Development and evaluation of a learning strategy program. *Journal of Educational Psychology, 71*, 64-73.
- Dillenbourg, P. (2002). Over-scripting CSCL: The risks of blending collaborative learning with instructional design. In P. A. Kirschner (Ed.), *Three worlds of CSCL. Can we support CSCL?* (pp. 61-91). Heerlen, NL: Open Universiteit Nederland.
- Dillenbourg, P., & Jermann, P. (in press). Designing integrative scripts. In F. Fischer, H. Mandl, J. Haake, & I. Kollar (Eds.), *Scripting computer-supported collaborative learning: Cognitive, computational and educational perspectives*. New York: Springer.
- Dillenbourg, P., & Tchounikine, P. (to appear). *Flexibility in macro-scripts for CSCL*. Submitted to International Conference of the Learning Sciences 2006.

- Doise, W., & Mugny, G. (1984). *The social development of the intellect*. Oxford: Pergamon Press.
- Hamilton, R. J. (1997). Effects of three types of elaboration on learning concepts from text. *Contemporary Educational Psychology, 22*(3), 299-318.
- Herring, S. C. (1999). Interactional coherence in CMC. *Journal of Computer-Mediated Communication, 4*(4).
- King, A. (1989). Effects of self-questioning training on college students' comprehension of lectures. *Contemporary Educational Psychology, 14*, 366-381.
- King, A. (1994). Guiding knowledge construction in the classroom: Effects of teaching children how to question and how to explain. *American Educational Research Journal, 30*, 338-368.
- Kollar, I., Fischer, F., & Hesse, F. (in press). Computer-supported cooperation scripts – A conceptual analysis. *Educational Psychology Review*.
- Kuhn, D. (1991). *The skills of argument*. Cambridge: Cambridge University Press.
- MOSIL (2004). *Framework for integrated learning*. Retrieved January 13th, 2006, from <http://www.iwm-kmrc.de/cossicle/resources/D23-05-01-F.pdf>
- O'Donnell, A. M. (1999). Structuring dyadic interaction through scripted cooperation. In A. M. O'Donnell & A. King (Eds.), *Cognitive perspectives on peer learning*. (pp. 179-196). Mahwah, NJ: Lawrence Erlbaum Associates.
- O'Donnell, A. M., & Dansereau, D. F. (1992). Scripted cooperation in student dyads: A method for analyzing and enhancing academic learning and performance. In R. Hertz-Lazarowitz & N. Miller (Eds.), *Interaction in cooperative groups: The theoretical anatomy of group learning*. (pp. 120-141). New York: Cambridge University Press.
- O'Donnell, A. M., Dansereau, D. F., Hall, R. H., & Rocklin, T. R. (1987). Cognitive, social/affective, and metacognitive outcomes of scripted cooperative learning. *Journal of Educational Psychology, 79*(4), 431-437.

- Piaget, J. (1985). *The equilibration of cognitive structures: The central problem of intellectual development* (T. Brown & K. J. Thampy, Trans). Chicago: University of Chicago Press.
- Schank, R. C., & Abelson, R. (1977). *Scripts, plans, goals, and understanding*. Hillsdale, NJ: Lawrence Earlbaum Associates.
- Slavin, R. E. (1983). When does cooperative learning increase student achievement? *Psychological Bulletin*, 94(3), 429-445.
- Van Boxtel, C., van der Linden, J., & Kanselaar, G. (2000). Collaborative learning tasks and the elaboration of conceptual knowledge. *Learning and Instruction*, 10, 311-330.
- Vygotsky, L. S. (1978). *Mind in society. The development of higher psychological processes*. Cambridge, MA: Harvard University Press.
- Webb, N. (1989). Peer interaction and learning in small groups. *International Journal of Educational Research*, 13(1), 21-39.
- Webb, N. M., & Palincsar, A. S. (1996). Group processes in the classroom. In D. C. Berliner & R. C. Calfee (Eds.), *Handbook of educational psychology* (pp. 841-873). New York: Macmillan.
- Weinberger, A., Ertl, B., Fischer, F., & Mandl, H. (2005). Epistemic and social scripts in computer-supported collaborative learning. *Instructional Science*, 33(1), 1-30.
- Weinberger, A., & Fischer, F. (2006). A framework to analyze argumentative knowledge construction in computer-supported collaborative learning. *Computers & Education*, 46, 71-95.

Appendix A

The “Social Script” (Weinberger et al., 2005)

Components

Resources & Participants: An equal amount of at least 2 case descriptions and participants.

Groups: Case groups.

Roles: An analyst and a critic.

Activities: ¹: applying theoretical concepts to cases and constructing arguments;
²: critiquing (in the specified sequence of eliciting clarification, identifying conflicts and constructing counter-arguments)

Group Formation & Task Distribution

For each case description one “case group” is formed, composed of all participants available. All case descriptions are distributed evenly amongst all case groups. Roles are distributed amongst all participants and amongst all groups in such a way that each participant assumes the role of analyst in one group and the role of critic in all other groups (cf. table A1).

Table A1. Role distribution in the social script. The number before each role signalizes the sequence in which these roles are to be assumed by the individual participant.

	<i>Group 1</i>	<i>Group 2</i>	<i>Group 3</i>
<i>Participant 1</i>	1. Analyst	2. Critic	3. Critic
<i>Participant 2</i>	3. Critic	1. Analyst	2. Critic
<i>Participant 3</i>	2. Critic	3. Critic	1. Analyst

Sequencing

Within each case group, ...

... the analyst writes a case analysis ¹

-- wait for all case group analysts to be done --

... each critic in turn writes a first critique of the case analysis²
-- wait for all case group critics to be completed --
... the analyst considers each critique and writes a reply to each in turn¹
-- wait for all case group analysts to be completed --
... each critic in turn reads the reply and writes a second critique²
-- wait for all case group critics to be completed --
... the analyst considers all critiques and writes a new case analysis¹

Regarding the sequencing principle: The roles of each participant rotate twice over the course of the script. Each participant begins with the role of analyst and changes his role to the first critic, second critic, etc. This cycle is repeated and ends with each participant resuming the role of analyst.

Appendix B

The “Universanté Script” (Dillenbourg & Jermann, in press)

Components

Resources: Case descriptions from at least two themes, with at least two case descriptions per theme.

Participants: Participants from at least two nations, with at least as many participants per country as there are case descriptions.

Groups: Theme groups, case groups and country groups.

Roles: none

Activities: ¹: analyzing and elaborating the case; ²: summarizing and explaining; ³: analyzing, comparing and relating new information to personal prior knowledge; ⁴: elaborating and explaining; ⁵: giving feedback and critiquing; ⁶: problem solving

Group Formation & Task Distribution

For each case description one “case group” is formed, composed of at least one participant per country, balanced amongst the groups. All case descriptions are distributed evenly amongst all case groups. For each theme, one “theme group” is formed, composed of all “case groups” with case descriptions centred on this theme.

Sequencing

Within each case group, all participants discuss the case ¹.

Within each country group, the members of each theme group in turn present a synthesis of their case experience ².

Within each theme group, the members of each country group create a fact sheet concerning the theme’s status within their country ².

Within each theme group, all participants discuss the similarities and differences between the fact sheets of different countries ³.

Within each country group, and for each theme group in turn, ...

... the members of the theme group present their fact sheet ⁴

... everybody else provides comments on the presented fact sheets ⁵

... the members of the theme group modify their fact sheet according to the comments ².

Within each case group, all participants propose a solution for the case problem ⁶.

Appendix C

The “ArgueGraph Script” (Dillenbourg & Jermann, in press)

Components

Resources: One copy of a questionnaire for each participant and another copy for each small group. The questionnaire consists of multiple-choice questions (all choices are equally valid) with space for providing an argument to justify the choice. One argument sheet per question of the questionnaire and as many copies of these sheets as are needed to provide one copy for each participant.

Participants: An even number of at least four participants (works best with 20-30 participants) and a tutor.

Groups: Class group and small groups.

Roles: none

Activities: ¹: justifying and generating arguments; ²: comparing, evaluating and elaborating; ³: negotiating and constructing arguments; ⁴: explaining and justifying; ⁵: summarizing and making connections

Group Formation & Task Distribution

In the “survey” phase, all participants together form the class group and receive one copy of the questionnaire. In the “conflict” and “elaboration” phase, all participants are distributed evenly into groups of two, composed of participants with maximal difference in their responses to the questionnaire. Each small group receives another copy of the questionnaire.

Sequencing

“Survey” phase:

Within the class group, all participants individually fill out the first copy of the questionnaire.¹

The tutor displays the aggregated results of the questionnaire (the participants' choices are plotted as anonymous points in the graph) to the participants.

Within the class group, all participants jointly discuss the displayed results of the questionnaire.²

“Conflict” phase:

Small groups are formed based on each participants' responses to the questionnaire.

Within each small group, all participants jointly fill out the second copy of the questionnaire, i.e., they negotiate on a single choice and generate a shared argument for this choice.³

“Elaboration” phase:

The tutor collects all questionnaires.

For each question of the questionnaire, the tutor compiles all choices and arguments from each small group on an argument sheet.

For each small group in turn, the tutor asks participants to comment on their arguments and gives advice on how to relate their arguments to theories and concepts.⁴

“Reflection” phase:

The tutor distributes all copies of the argument sheets to all participants.

Within the class group, each participant individually writes a synthesis of all arguments on the argument sheet, taking into account the advice of the tutor.⁵

Appendix D

The “MURDER Script” (Dansereau et al., 1979)

Components

Resources: Text material with a small number of text passages for each small group.

Participants: An even number of at least four participants.

Groups: Pairs.

Roles: A summarizer and a listener.

Activities: ¹: relaxing, focusing; ²: reading, monitoring comprehension; ³: summarizing, explaining; ⁴: monitoring, giving feedback; ⁵: elaborating; ⁶: reviewing, reflecting

Group Formation & Task Distribution

All participants are grouped into pairs. All text material is distributed amongst the pairs.

Within each pair, one participant assumes the role of summarizer and the other the role of listener.

Sequencing

Within each pair and for each passage of text, ...

... both participants set the *Mood* for studying ¹.

... both participants read the text material for *Understanding* ².

... the summarizer *Recalls* the material ³.

... the listener *Detects* errors/omissions and gives feedback ⁴.

... both participants *Elaborate* on the learning material ⁵.

... both participants *Review* the learning material and what they have learned ⁶.

-- after each cycle, participants rotate roles --