Scripting and Orchestration: Recent Theoretical Advances

Organizers
Frank Fischer, Ludwig Maximilian University, Munich, Germany Frank.Fischer@psy.lmu.de
Jim Slotta, University of Toronto, Toronto, Canada. jslotta@gmail.com

Discussant
Clark Chinn, Rutgers University, clark.chinn@gse.rutgers.edu

Abstract: This symposium brings together four research groups that have been working on advancing theoretical models of guidance for CSCL. The models share the emphasis on scripting and orchestration but vary in terms of their specific focus, grain size of collaboration, and the nature of learning activities they address. The goals of this symposium are, first, to present recent advances in theorizing guidance from a scripting and orchestration perspective, and, second, to discuss commonalities, differences and future trajectories for theory development on guidance for CSCL. The audience will be interactively involved by using technologies for knowledge building. These technologies allow the audience to contribute challenging cases, questions, and ideas for studies during and after the presentations via laptops and smart phones. To approach the second goal, the discussant will draw from these contributions and involve the audience and the presenters in refining and synthesizing the ideas in a final synthesis discussion.

Introduction

In computer-supported collaborative learning, learners are often working together on complex problems requiring them to conduct inquiries and design complex artifacts. There is a stockpile of empirical evidence that activities such as problem-solving, inquiry and design support deep learning, particularly when the learner is scaffolded (Hmelo, Holton & Kolodner, 2000; Linn & Eylon, 2006; Quintana, Reiser, Davis et al., 2004; Schauble, Glaser, Schulze & John, 1995). Without guidance, many learning designs would transpire in a way that diverges from the design, making it difficult to test theoretical conjectures, or build constructive models. Guidance can be provided by different sources (e.g., teacher, computer, peer), on different social levels (individuals in groups, small groups, class, communities) and with different types of scaffolding (channeling, prompting, hinting, etc). There is now an impressive number of conceptual and empirical papers on the topic of guidance in CSCL. Still, apart from the grand theories of Vygotskij and Piaget, there has been limited progress in the development of new theories and theoretical models of guidance in CSCL (i.e., to enable and support collaborative learning).

In this symposium, we bring together 4 research groups that have been working intensively on developing theories or theoretical models of guidance for CSCL. The work from these groups varies in terms of its grain size of collaboration, the nature of learning activities it investigates, and to the extent to which they explicitly conceptualize the role of the teacher. Indeed, not all contributors would say they are explicitly advancing a "theory," although all are advancing explicit theoretical positions in the form of models or frameworks. The goals of this symposium are (1) to present recent advances in theorizing guidance for CSCL from a scripting and orchestration perspective, (2) to discuss commonalities and differences between the approaches and future trajectories for theory development on guidance for CSCL.

Overview of contributions

The symposium includes 4 contributions, each elaborating on one theoretical approach of scripting. They have in common that they are aiming at explaining and improving guidance for CSCL. The Dillenbourg paper on orchestration offers a historical perspective on instructional guidance in CSCL that identify the intellectual roots of the research field to help locating the current approaches to scripting and orchestration in a bigger picture. Moreover, the paper sketches a model of orchestration in which instructional guidance is seen as an organic and multi-level process to optimize the classroom beyond cognitive learning, with respect to multiple constraints that exist in real world classrooms. In this process, external collaboration scripts are only one of several aspects and the teacher plays an eminent role in aligning the aspects before and during the lesson. The paper by Fischer et al. on the script theory of guidance addresses how internal collaboration scripts (i.e. knowledge on collaboration) develop through participation in CSCL practices, and how these practices are, in turn, shaped by the internal collaboration scripts of the participating group members. With respect to guidance, external collaboration scripts (i.e. instructional support for collaboration) are seen as a set of scaffolds facilitating participation in CSCL practices through activation of internal collaboration script components that would not have been spontaneously transferred into the CSCL practice. The Tchounikine paper on appropriation focuses on the dangers but also the chances that come with the phenomenon that learners are not interacting with external scripts and tools as they are designed and intended by the designer but as the learner understands them and as they increasingly

integrate them in their existing cognitive tool kit, based on their goals and on their prior experiences. The Slotta paper on the knowledge community and inquiry model has its main focus on the dynamic interplay of individual cognition and a developing collective knowledge base. With respect to instructional guidance, he advances a scripting perspective to connect the individual and small group level to the level of collective epistemic activities.

Our discussant, Clark Chinn, will identify overarching issues and important lines of future research in CSCL, presenting an interpretation of the presented approaches.

Interaction with audience

An interesting challenge for the CSCL conference, posed by Jeremy Roschelle, Roy Pea and others, is to show better progress within our conference meeting of the kinds of interactions we would advocate for collaborative learning in our research. Over time, our community should offer a wider range of experiences to the conference audience (both those who attend and those who may need to participate from elsewhere). Using "computer supported" technologies -- from audience response systems to wikis, to ubiquitous computing and augmented reality -- we should enable real time contributions and rich discussions or "knowledge work" to be performed. This is not an easy transition for a research community to make, particularly when it involves new forms of interaction that are only beginning to solidify. However, if there is any community that should rightfully explore such interactions it is CSCL. To that end, we will integrate a new form of knowledge building technology developed by Slotta and his team at the University of Toronto, to support audience engagement and dynamic discussion. Anyone in the audience with a laptop, smart phone or tablet (iPad, etc) will be able to log on during and after the presentations and contribute challenging cases, questions, and ideas for further research. The discussant will draw from these contributions and involve the audience and the presenters in refining the ideas in a final synthesis discussion.

Paper 1. Mediating, Scripting, Orchestrating: the Evolution of Instructional Design in CSCL

Pierre Dillenbourg, Ecole Polytechnique Federale de Lausanne, Switzerland, pierre.dillenbourg@epfl.ch

We rarely use the terms "instructional design" in CSCL, even though much of our work is about designing learning activities. I argue that, through its evolution, CSCL enriched instructional design with several concepts. I illustrate this evolution with the design of argumentation activities.

The foundational idea of CSCL is actually that collaboration can be 'designed' (Roschelle, 1990): communication tools do not simply transport messages between learners but shape their interactions. This mediation effect is obvious when learners seat in front of the same computer, i.e. when computers are not justified by data communication. For instance, CSCL scholars designed graphical editors providing learners with argumentation primitives such as 'hypothesis', 'evidence', etc. (Suthers et al, 1995). The design of these tools indeed constitutes a piece of instructional design since the interactions induced by the interface are expected to trigger specific cognitive processes, which is the craft of instructional design. This type of design nonetheless constitutes an expansion of what was understood as instructional design in the eighties; it is a more subtle or indirect way to induce targeted interactions, closer to the way 'design' is understood outside education. In the next decade, CSCL produced stronger forms of interactions shaping, namely 'scripts': the learners had to follow a sequence of interactions steps considered as - to reuse the same example- necessary for a well-formed argumentation (Weinberger et al, 2002). If Suzanne produces a claim, John would be prompted to provide counter-evidence. This approach is closer to the constraining flavor of instructional design but brings a novelty: the script is designed to be internalized by learners, which is not the case for the lesson plans produced by instructional design.

In the same decade, a third approach emerged. For triggering argumentation, one can ask peers to read texts that would provide them with conflicting evidence ('jigsaw' method). Alternatively, one may identify learners who have opposite opinions and then pair them for argumentation activities. These pedagogical scenarios are closer to what is produced by instructional design (Jermann & Dillenbourg, 1999). They nonetheless expand instructional design by integrating individual, collaborative and class-wide activities into a consistent workflow. Confusingly, these methods were also called 'scripts'. We renamed macro-scripts or classroom scripts, while the scripts-to-be-internalized were renamed 'micro-scripts'.

In the last decade, micro- and macro- scripts left research labs, they did not enter into classrooms without difficulty. The need for flexibility emerged as a key issue: since scripts introduce structure in teamwork, it may occur that this structure is too rigid or inappropriate and has to be changed on the fly. Adaptation has always been a central concept in instructional design, usually referring to pre-defined changes of learning activities based on learners' behavior. In 'adaptive instruction', the design process includes the design of what can be changed during the enactment of the lesson plan. In CSCL, the need for adaptation cannot completely be anticipated. Macro-scripts are not closed environments; they stretch over multiple activities and tools (Dillenbourg &

Hong, 2008). Adaptations may be required by any event that populates the classroom ecosystem: students arriving late, discipline problems, lack of time, … Therefore, the term 'flexibility' as a different flavor that 'adaptation': it is about empowering users, teachers and learners, in changing themselves the lesson plan. Flexibility is nonetheless an instructional design concept: since workflows are per definition rigid structures, new forms of flexibility had to be invented such as 'team jokers' or 'orchestration cards' (Dillenbourg & Tchounikine, 2007). Flexibility is still a key challenge for instructional design, both technically and pedagogically.

During this last decade, a less visible but more fundamental change has appeared. It concerns the result of instructional design. In school-oriented CSCL research, what is designed is not anymore a single activity or a piece of software but a richer set of activities or tools to be orchestrated by a teacher. In this case, instructional design includes designing the ways in which the teachers will handle complexity. The terms 'design for classroom orchestration' (Dillenbourg, to appear) include design concerns that were not salient before in instructional design such as minimizing the cognitive load induced by monitoring the learners, facilitating what has traditionally been referred to as 'classroom management', maximizing curriculum relevance, optimizing teaching time and teacher's energy, etc. This vision stretches instructional design over new concerns such the physicality of the classroom and of the hardware. It pays attention to elements that what would in the past be disregarded as belonging to the logistics of education (e.g. time to move tables). This broader view of design constitutes a contribution from CSCL to instructional design (if one is does not understand 'instructional' in a narrow sense). What is designed is not a single entity with intrinsic educational affordances. The design produces a set of elements that integrate into classroom to form a distributed system. In this organic view of design, the goal is not only to maximize learning gains but also to satisfy classroom constraints (time, discipline, curriculum,…) with a minimal orchestration load.

Orchestration hence addresses at guidance from a complementary angle; it is not about how much guidance or structure the teams need, but how much guidance the teacher is able provide, given his or her heavy load. It takes into consideration My 'petite theory' of orchestration includes two models. The first model describes pedagogical scenarios or scripts as multi-plane and multi-layer activity graphs. These vertices of the graph are the learning activities. The term "multi-plane" describe the fact that these activities may occur at six levels, named "planes" by Vygostky: individual, team, class, periphery, community and world. The edges of the graph are links between activities. The weight of a link [ai aj] is the probability that an aj can be conducted despite the fact that learners have not completed aj. Examples of edges are dataflow, prerequisites, motivation didactic elicitation, … The flexibility of a graph depends of the link weights and can be defined as the effort necessary to transform an activity graph on the fly. A notation system has been developed to represent these graphs (Figure 1).

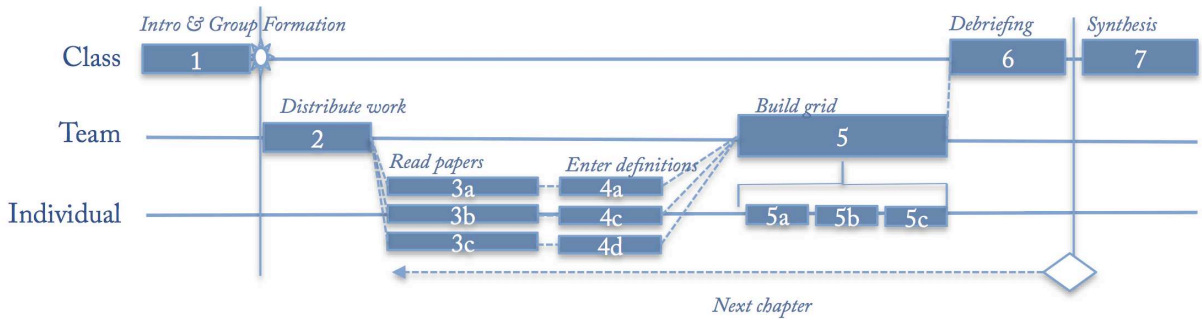

Figure 1. Representing the activity graph of a macro-script

The term 'multi-layer' refers to the fact that graph does not only exist as a digital structure but also as physical structure. I will present some paper-based interfaces in which the teacher may adapt the graph by handling paper cards. The great flexibility of paper reminds the study from Hutchins on 'how a cockpit remember its speed'. Surprisingly, few authors reuse his notion of distributed systems to model a classroom, while it has a great potential.

The second model borrows ideas from information theory in order to the flows of information in a classroom. For instance, when a teacher visually scans the faces of the students in the classroom, he builds a representation of their attention level, with a certain degree of uncertainty. If he or she writes on the backboard for 20 seconds, he does not update his model and the uncertainty increases. Entropy does only decrease without external energy and this energy mostly come from the teacher, who will ask for instance new questions to reduce the uncertainty of his model. This barely legal reuse of information theory inspires the design of new type of learning technologies, namely buffers, for reducing the information load that emerged from monitoring and scaffolding learners.

Paper 2. Advances in the Development of a Script Theory of Guidance for Computer-Supported Collaborative Learning

Frank Fischer, Ingo Kollar, Karsten Stegmann & Christof Wecker, Ludwig Maximilian University, Munich, Germany, frank.fischer@psy.lmu.de

In this contribution, we will present an outline of a script theory of guidance for computer-supported collaborative learning that builds on the basic distinction of internal scripts (understood as individual memory structures) and external scripts (as instructional interventions that structure collaborative learning processes). The script theory of guidance builds on two quite distinct theoretical traditions. First, it refers to recent schema-theoretic accounts that have broken down the initially quite rigid script concept into multiple components of a dynamic memory. These components are considered to be flexibly combined in response to characteristics of the situation and the individual´s goals (e.g., Kintsch,1998; Schank, 1999). Second, it builds on socio-cultural theory with its assumptions on the relation of discourse on the social plane and the development of complex cognitive skills (e.g., Vygotsky, 1978).

In the script theory of guidance, internal collaboration scripts represent knowledge on collaboration that enable learners to effectively understand and act in recurring CSCL practices (Kollar, Fischer & Hesse, 2006). CSCL practices are, in turn, shaped (but not determined) by the internal collaboration scripts of the participating individuals. In recent conceptions of the theory internal collaboration scripts are considered to be composed of the hierarchically organized components play, scenes, roles and scriptlets (Fischer, Kollar, Stegmann & Wecker, 2013). Internal collaboration scripts may change dynamically if they do not lead to successful understanding or acting in a CSCL practice. New internal collaboration script components develop as re-configurations of existing components that have been instrumental in reaching the learner's goals.

External collaboration scripts consist of scaffolds that stimulate functional or inhibit dysfunctional internal script components (Fischer et al., in press): play scaffolds, scene scaffolds, role scaffolds, scriptlet scaffolds that are designed to support the learning of internal script components and of subject matter knowledge. As a basic tenet of the theory, internal script components that already exist in the learners' memory need to be taken into account in order to foster collaboration and learning in CSCL. The script theory of guidance for CSCL can inform the design of external collaboration scripts on when external collaboration scripts can effectively be used, which type of scaffold is likely to support learners in employing functional internal script components, and how these scaffolds can be faded to increase a self-directed configuration and re-configuration of internal script components (e.g., Wecker & Fischer, 2011). The script theory of guidance for CSCL specifies seven principles on (1) how CSCL practices are shaped through situational constraints and the internal collaboration scripts of the participating learners, (2) how internal collaboration scripts develop through participation in these CSCL practices, and (3) how external collaboration scripts can support the development and application of internal collaboration script components through scaffolding of different component levels. For example, the transactivity principle states that the more a given CSCL practice requires the transactive application of knowledge, the better this knowledge is learned through participation in this CSCL practice. The optimal scripting level principle states that external collaboration scripts should have the largest effects on learning if the scaffolding is targeted at the highest hierarchical level of the internal collaboration script (e.g., the play), where components on the subordinate levels (e.g., scenes, scriptlets) are already available to the learner. Empirical studies in CSCL provide evidence for the script theory of guidance. For example, they show that well-designed external collaboration scripts enable learners to engage in CSCL practices on a level beyond what they would be able to achieve spontaneously (e.g., Schellens, De Wever, van Kehr & Valcke, 2007; Schoonenboom, 2008). Other studies found evidence for example supporting the transactivity principle (e.g., Stegmann, Weinberger & Fischer, 2011). However, more specific investigations on the validity of the seven principles are needed.

Paper 3. Towards Theoretical Bases to Acknowledge Appropriation Phenomena in CSCL

Pierre Tchounikine, Université Joseph Fourier, Grenoble, France, pierre.tchounikine@imag.fr

Many CSCL settings are based on the hypothesis that the technological enactment framework provided to students will have an influence on their behavioral and cognitive activity. This is particularly the case for scripting approaches, within which the instructional design is reified by both the script (the instructions) and the technology (the system interface or data/work-flow).

However, the activity that will emerge from the confrontation of students with the task and the technological setting may be influenced by different dimensions, and is subject to different contingences (Tchounikine 2008, Tchounikine 2011). Teachers set tasks and students interpret the specifications of the task, their subsequent activity being a more or less rational response to the task (Goodyear 2001). Activity is related to the task but also to other dimensions (e.g., students' effective motivations or perception that is developed by the students

of the script and the provided technological setting) that evolve in time, and are interrelated within systemic relations. Typically, technical functionalities and properties are not received raw but actively constituted on receipt by the user (Jones et al. 2006). The characteristics of the technological setting will be interpreted in different ways by learners, who will appropriate them, in context, according to their purposes and in terms of their own current interests or needs. Unexpected usages of educational software are frequent and are not a matter of "good" or "bad" design only (as examples: usage of chat as a means of perception for mutual presence or actions; usage of a function meant to edit a result as a "support for thinking" or, vice versa, editing of a result (elaborated via other means) with tools meant to elaborate the result, thought of and considered as a "support for thinking" and a vector for the targeted learning; change in the way the environment is used due to the evolution of motivations (and, thus, effective activity), for example from "playing the game of the pedagogic contract and using the platform to meet the teacher's demand" to "deal with urgency and produce the expected result (whatever the means are)". Basically, learners do not use the provided software to solve the set task, they consider some task (that may only correspond more or less to that set by the teacher) and take advantage of the means that seem best adapted to them (which may correspond to some usage of the provided software) in the context of their activity.

In contexts within which the learning setting designed artifacts (the script, the technological enactment framework) are supposed to influence the students' activity, the fact they may be appropriated in different ways is of importance. CSCL cannot escape the theoretical issue of understanding how actors (teachers, students) appropriate these artifacts.

In line with activity-oriented works (Kaptelinin & Nardi 2006), it may be hypothesized that the constructive nature of activity impacts users' use and appropriation of software and scripts. Designers create artifacts based on how they imagine their future usage, yet artifacts only become instruments for users in the context of these users' activities, i.e. when and through the way they allow these users to achieve the tasks they consider in the way they consider them. When designing artifacts, we need to take care that when a user adapts an artifact as a way to adapt it to his/her activity, this adaptation may present a situated dimension, an interpretative dimension, a constructive dimension and/or an oriented dimension.

In CSCL settings the script, the teachers' activity and the technologies (when designed as flexible technologies (Dillenbourg & Tchounikine 2007, Sobreira & Tchounikine 2012) may be adapted. The way the script and/or the technical framework may be adapted to comply with appropriation issues while remaining coherent with the script's design rationale (i.e., why and how it is meant to support learning) must be studied in relation with an understanding of the scripting principles and effects. Moreover, adaptation issues must also be studied in relation with how teachers conducting the session may be empowered to understand and orchestrate the script's enactment. The development of a theoretical perspective to appropriation must thus be conducted in relation with theoretical developments related to scripting and orchestration.

Paper 4. The Knowledge Community and Inquiry Model: Scaffolding individual, Collaborative, and Collective Activities

Jim Slotta, University of Toronto, 252 Bloor St. West, Toronto, Canada, jslotta@gmail.com

This paper advances Knowledge Community and Inquiry (KCI) as a pedagogical model that guides the design of complex inquiry curriculum that includes individual, collaborative (i.e., small group) and collective (whole class or multiple class) activities. In KCI (see Figure below), all individuals and groups work together to produce a collective knowledge base that serves as a resource for subsequent inquiry activities. The curriculum is several weeks or months in duration, and includes technology-enhanced materials, tools, and virtual environments that scaffold the various designed interactions.

In contrast with other "knowledge community" approaches (e.g., Scardamalia & Bereiter, 2006), KCI includes scripted inquiry that is carefully designed to address the science learning goals. Thus, while KCI is theoretically committed to the notion of collective epistemology, it has one foot planted firmly in the theoretical space of scaffolded inquiry (Slotta & Peters, 2008). The model is inspired partly by "Web 2.0" environments, including content communities (e.g., Wikipedia, YouTube) and social networks (Facebook), which are increasingly familiar to students, teachers and researchers (Slotta & Najafi, 2012, Figure 2). So, while KCI attempts to get students working together ("one for all"), it also makes the advantages of that collective achievement accessible to individual or small group inquiry ("all for one").

KCI curriculum is developed through a sustained co-design effort including teachers, researchers, technology developers, and interaction designers. The designed artifact constitutes a "script" that includes real-time decisions or assignments made by intelligent agents, and involves student-contributed content, social tagging, learning in ubiquitous and distributed contexts, and a wide range of individual and collaborative (i.e., small group) scripted activities. The script must be "orchestrated" by the teacher, who is greatly enabled by the technology-environment. Below, I describe a set of design principles that guide our creation of individual, collaborative, cooperative and collective inquiry activities, and how those activities are scaffolded.

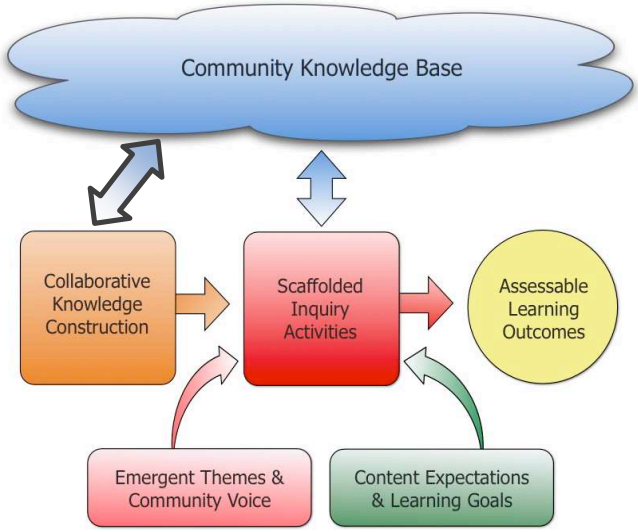

Figure 2. Knowledge Community and Inquiry Model: Basic Processes and Constraints

Principle 1. Students work collectively as a knowledge community, creating a knowledge base that serves as a resource for their ongoing inquiry within a specific science domain. Students are scaffolded to work collectively (i.e., all students working in parallel, building on one another's contributions, as in wiki editing). It is not an ill-defined task, however, for students to build any knowledge base that appeals to them (i.e., as in discovery oriented inquiry); rather, the science content expectations are used as an explicit framework or index that scaffolds the collaborative construction of a relevant and accessible knowledge base. In a recent curriculum on climate change, we established a knowledge base in the form of a wiki, with blank page "templates," where each new page would be a major climate change issue. Students were responsible for coming up with the issues, but each issue page included 5 sub-headers that were not negotiable, corresponding to the 5 scientific concepts of the curriculum unit: carbon sinks/sources; greenhouse gases; energy currents in ocean and atmosphere; scientific models and forecasts; and remediation efforts

Principle 2. The knowledge base is accessible for use as a resource as well as for editing and improvement by all members. In the climate change unit described above, students came to understand that the knowledge base could serve as a collectively constructed and validated resource for their community. Notably, 5 class sections (n= 121) of a high school biology course collaborated in creating a single common wiki, using our blank template pages. The result was 14 major issue pages, each averaging 3509 words, with 90 page revisions made by 8 authors. This is a rarely observed level of collaborative knowledge production, and demonstrated to researchers and teachers alike that students enjoyed the task and excelled at it.

Principle 3. Collaborative Inquiry activities are designed to address the targeted science learning goals, including assessable outcomes. Inquiry is seen as a process where individual learners build on their existing ideas to develop scientific understandings (e.g., Linn & Eylon, 2006). Students work individually and in small groups, making use of the community knowledge base as a key resource. We designed a "climate change remediation" assignment, where groups (size n=2, 3 or 4) collaboratively designed a remediation that addressed at least 3 of the issues in the knowledge base – targeting specific geographical regions of Canada. One requirement was to make explicit connections to each of the 5 major scientific elements listed above (ie, assessable outcomes). Just as the collaborative knowledge construction phase was scaffolded through the use of wiki page templates, this small group inquiry project was carefully scaffolded through the design of structured page templates and instructions (e.g., "make sure that you refer to at least 3 of the biodiversity issues".

Principle 4. The teacher's role must be clearly specified within the inquiry script, but also include a general orchestration role. In KCI, the teacher's role is that of an expert collaborator or mentor, responding to student ideas as they emerge, and orchestrating the pedagogical flow of activities. Teachers are not just a vague "guide on the side" – an image or assignment that often paralyzes teachers or leaves them "sidestreamed" by the overly scaffolded learning environments, which often include no explicit role for the teacher. KCI curriculum includes specific, scripted interactions with students or responses to materials, such as providing feedback and making "consequential" orchestrational decisions based on the content of student interactions and artifacts. For example, in a recent physics activity, we engaged students in solving ill structured problems in a smart classroom setting, where they worked collaboratively at various stations within the room (e.g., inputting their tags and votes on personal tablet computers, then using collaborative Smartboard (i.e., interactive whiteboard) activities to negotiate a consensus). After setting up the problems for solution, the group "submitted" their readiness, and the teacher's his own tablet) was notified that a group was ready for a debriefing encounter. The teacher then walked over to that group's station and consulted with them about their problem set-up. If (and only if) they had done a sufficiently detailed and accurate job, the teacher touched a "go ahead" button in his own tablet, and the student tablets were all refreshed with new tasks for the next scaffolded activity in the script.

The final example above illustrates the current focus of our research, with regard to scaffolding technologies as well as inquiry designs. Through painstaking design (which can take up to a year's time) we develop substantive curricula – typically whole semester or multi-week courses, carefully developing the script according to the KCI principles above. Once we have the pedagogical script, we design and develop all materials and scaffolding for activities at the individual, small group and whole class levels. For example, a student's personal tablet computer (e.g., iPad) could guide him or her to the appropriate location in the room, or to a particular small group, and then solicit, guide and collect particular observations or reflections. Internet-based software collects all student contributions, such that intelligent agents can perform real-time data mining to compile an aggregated dataset which can be presented on tablets or Smartboards, or both. Students respond to this emergent view, perhaps recognizing cells in a table where they disagree or need more data. Teachers can be prompted on their own personal tablets for a variety of interactions. Through such intricately designed and scaffolded activities, we seek to create a sense of autonomy, creativity and inquiry, without "overscripting."

References

Dillenbourg, P. & Tchounikine, P. (2007). *Flexibility in macro-scripts for CSCL. Journal of Computer Assisted Learning, 23*(1), 1-13.

Dillenbourg, P., & Hong, F. (2008). The mechanics of CSCL macro scripts. *International Journal of Computer-Supported Collaborative Learning*. 3 (1), pp. 5-23

Dillenbourg, P., & Tchounikine, P. (2007). Flexibility in macro CSCL scripts. *Journal of Computer Assisted Learning.* Vol. 23 (1), pp. 1-13.

Dillenbourg, P., Jarvela, S. & Fischer, F (2009). The evolution of research on computer-supported collaborative learning: from design to orchestration. In N. Balacheff, S. Ludvigsen, T. de Jong, A. Lazonder & S. Barnes (Eds) *Technology enhanced learning: Principles and Products.* (pp. 3-20). Springer.

Dillenbourg, P. (to appear) Design for Classroom Orchestration . Computers & Education

Fischer, F., Kollar, I., Stegmann, K. & Wecker, C. (2013). Toward a script theory of guidance in computer-supported collaborative learning. *Educational Psychologist, 48*(1), 56-66.

Goodyear, P. (2001). Effective networked learning in higher education: notes and guidelines. Volume 3 of the Final Report to JCALT: Networked Learning in Higher Education Project. Retrieved from http://csalt.lancs.ac.uk/jisc/ (December 2nd, 2006).

Jermann, P., & Dillenbourg, P. (1999). An analysis of learner arguments in a collective learning environment. C. Hoadley et J. Roschelle (Eds.), *Proceedings of 3rd Computer-Supported Collaborative Learning Conference*, pp. 265-273, Stanford .

Jones, C., Dirckinck-Holmfeld, L. & Lindström, B. (2006). A relational, indirect, meso-level approach to CSCL design in the next decade. *International Journal of Computer-Supported Collaborative Learning (1),* 35-56.

Hmelo, C.E., Holton, D.L., Kolodner, J.L. (2000). Designing to Learn about Complex Systems. *Journal of the Learning Sciences, 9 (3)*, 47-298.

Kaptelinin, V. & Nardi, B. (2006). *Acting with Technology: Activity Theory and Interaction Design*. Cambridge: MIT Press.

Kintsch, W. (1998). *Comprehension: A paradigm for cognition*. Cambridge: Cambridge University Press.

Kollar, I., Fischer, F., & Hesse, F. W. (2006). Collaboration scripts - a conceptual analysis. *Educational Psychology Review, 18*(2), 159-185.

Linn, M. C., & Eylon, B.-S. (2006). Science Education: Integrating Views of Learning and Instruction. In P. A. Alexander & P. H. Winne (Eds.), *Handbook of Educational Psychology* (2nd ed., pp. 511-544). Mahwah, NJ: Lawrence Erlbaum Associates.

Quintana, C., Reiser, B.J., Davis, E.A., Krajcik, J., Fretz, E., Duncan, R., Kyza, E., Edelson, D., & Soloway, E. (2004). "A scaffolding design framework for software to support science inquiry". *Journal of the Learning Sciences*, 13(3), 337-386.

Roschelle, J. (1990) *Designing for Conversations.* Paper presented at the AAAI Symposium on Knowledge-Based Environments for Learning and Teaching, Standford,CA, March 90.

Scardamalia, M., & Bereiter, C. (2006). Knowledge building: Theory, pedagogy, and technology. In K. Sawyer (Ed.), *Cambridge Handbook of the Learning Sciences* (pp. 97-118). New York, NY: Cambridge University Press.

Schank, R. C. (1999). *Dynamic Memory Revisited*. Cambridge, MA: Cambridge University Press.

Schauble, L., Glaser, R., Duschl, R. A., Schulze, S., & John, J. (1995). Students' understanding of the objectives and procedures of experimentation in the science classroom. *Journal of the Learning Sciences, 4*, 131-166.

Schellens, T., Van Keer, H., De Wever, B., & Valcke, M. (2007). Scripting by assigning roles: Does it improve knowledge construction in asynchronous discussion groups? *International Journal of Computer-Supported Collaborative Learning*, *2*(2-3), 225-246.

Schoonenboom, J. (2008). The effect of a script and a structured interface in grounding discussions. *International Journal of Computer-Supported Collaborative Learning, 3,* 327-341.

Slotta, J. D., & Peters, V. L. (2008). A blended model for knowledge communities. Proceedings of the Eighth International Conference for the Learning Sciences – ICLS 2008 (pp. 343-350).

Slotta, J.D. & Najafi, H. (2012). Supporting Collaborative Knowledge Construction with Web 2.0 Technologies. In *Emerging Technologies for the Classroom: A Learning Sciences Perspective* (N. Lavigne, Ed.).

Sobreira, P. & Tchounikine, P. (2012) A model for flexibly editing CSCL scripts. *International Journal of Computer-Supported Collaborative Learning 7* (4).

Stegmann, K., Weinberger, A., & Fischer, F. (2011). Aktives Lernen durch Argumentieren: Evidenz für das Modell der argumentativen Wissenskonstruktion in Online-Diskussionen {Active Learning Through Argumentation: Evidence for a Model of Argumentative Knowledge Construction in Online Discussions}. *Unterrichtswissenschaft, 39* (3), 231–244.

Suthers, D., Weiner, A. Connelly J. & Paolucci, M. (1995) Belvedere: Engaging students in critical discussion of science and public policy issues. In. J. Greer(Ed). Proceedings of the International Conference in Artificial Intelligence in Education, Washington, August 16-19, pp. 266-273.

Tchounikine, P. (2011). *Computer Science and Educational Software Design. A resource for multidisciplinary work in Technology Enhanced Learning.* New York: Springer.

Tchounikine, P. (2008). Operationalizing macro-scripts in CSCL technological settings. *International Journal of Computer-Supported Collaborative Learning 3(2),* 193-233.

Vygotsky, L. S. (1978). *Mind in society: The development of higher psychological functions*. Cambridge: Harvard University Press.

Wecker, C. & Fischer, F. (2011). From guided to self-regulated performance of domain-general skills: The role of peer monitoring during the fading of instructional scripts. *Learning and Instruction*, *21* (6), 746-756.

Weinberger, A., Stegmann, K., & Fischer, F. (2010). Learning to argue online: Scripted groups surpass individuals (unscripted groups do not). *Computers in Human Behavior, 26(4),* 506-515.

Weinberger, A., Fischer, F. & Mandl, H. (2002). Fostering computer supported collaborative learning with cooperation scripts and scaffolds. In G. Stahl (Ed.), *Computer support for collaborative learning: foundations for a CSCL community* (pp. 573-574). Proceedings of the Conference on Computer Support for Collaborative Learning (CSCL) 2002, Boulder, CO.