Classroom Orchestration: The Third Circle of Usability

Pierre Dillenbourg, Guillaume Zufferey, Hamed Alavi, Patrick Jermann, Son Do-Lenh, Quentin Bonnard, Sébastien Cuendet, Frédéric Kaplan, CRAFT, Ecole Polytechnique Fédérale de Lausanne (EPFL)
Email: Pierre.Dillenbourg@epfl.ch, Guillaume.Zufferey@epfl.ch, Hamed.Alavi@epfl.ch, Patrick.Jermann@epfl.ch, Son.Dolenh@epfl.ch, Quentin.Bonnard@epfl.ch, Sebastien.Cuendet@epfl.ch, Frederic.Kaplan@epfl.ch

Abstract: We analyze classroom orchestration as a question of usability in which the classroom is the user. Our experiments revealed design features that reduce the global orchestration load. According to our studies in vocational schools, paper-based interfaces have the potential of making educational workflows tangible, i.e. both visible and manipulable. Our studies in university classes converge on minimalism: they reveal the effectiveness o tools that make visible what is invisible but do not analyze, predict or decide for teachers. These studies revealed a third circle of usability. The first circle concerns individual usability (HCI). The second circle is about design for teams (CSCL/CSCW). The third circle raises design choices that impart visibility, reification and minimalism on classroom orchestration. The fact that a CSCL environment allows or not students to look at what the next team is doing (e.g. tabletops versus desktops) illustrates the third circle issues that are important for orchestration.

Introduction

When CSCL was implemented on desktops, the physical organization of classrooms was rarely addressed as a research topic. The use of small mobile devices as well as large immobile devices (e.g. tabletops) brought forward the physical orchestration of the classroom. An early example was given by Roschelle and Pea (2002): when a student walks across the classroom to share PDA data by infrared instead of sending them wirelessly, this publicly visible walk provides the teacher and other students with the awareness of the actual dataflow. Making the educational workflow visible and tangible has an effect on classroom orchestration. A recent example comes from Nussbaum et al (2010). All kids in a classroom interact with a mouse on a single display. Each student owns a tiny subset of the display area, as small as a phone display. The very same learning activity could be conducted on a central display or on personal displays yet leads to completely different forms of orchestration. Analyzing the impact of CSCL design on classroom orchestration is the goal of this conceptual paper. Therefore, we first present a study where we used an augmented reality simulation in classrooms for two years. This simulation combines a tangible interface and a paper-based interface. The added value of tangibles was obvious in terms of individual usability and teamwork, but we also understood that the added value of paper concerned a third circle of usability: its integration in the classroom ecosystem. While paper-based computing is a good example of orchestration, we aim to go higher in abstraction. Therefore, in the second section, we define this third circle with concepts of orchestration, awareness and workflow. They are illustrated with new designs and new results in the last sections with paper computing as well as with other technologies.

Why is Paper Highly Usable in Classrooms?

The spread of reading devices led scholars to compare digital and paper documents in terms of readability or annotations. We analyzed paper interfaces from another perspective: classroom orchestration. Tinker is an augmented-reality simulation for training apprentices in logistics (Zufferey et al, 2009). Teams build the mock-up of a warehouse by placing shelves on a table (Fig. 1 left). The TinkerLamp includes a camera and a beamer. It recognizes the visual markers on the shelves, computes a model of the warehouse and displays information on the table and on the shelves (their contents, the movement of the forklifts, the surfaces used, etc.).

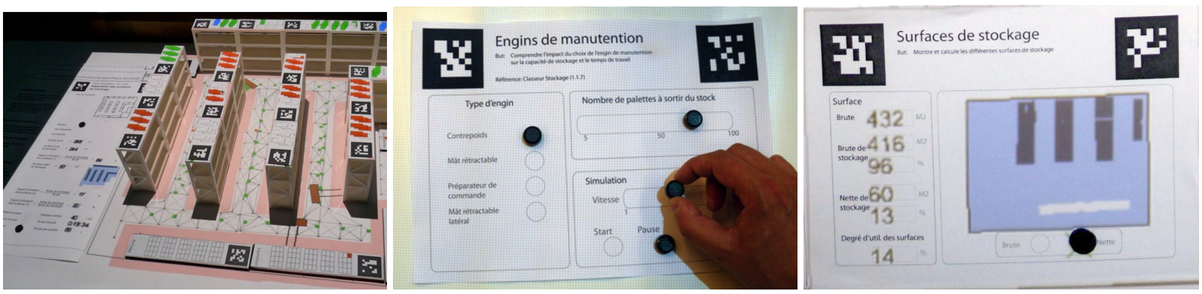

Figure 1. Input (Middle) and Output (Right) Sheets for a Tangible Simulation (Left).

Empirical studies revealed that this tangible interface outperformed a multi-touch table running the same simulation (Lucchi et al, 2010). Tinker has been used in 10 classes from 5 schools, through several studies, but in the beginning, teachers did not feel comfortable in setting up the activities and controlling the simulation. After several prototypes, we found that interactive paper sheets facilitated their interactions. Teachers and apprentices interact with the system by placing tokens on the sheets (Fig.1 middle and right). The camera identifies the sheet and its orientation thanks to the markers, retrieves the coordinates of the input area and identifies the black tokens as equivalent to mouse clicks. On these sheets, the system beams information such as performance measures (e.g. the average time to bring a box from the shelves to the truck) and a reduced map of the warehouse. These paper sheets are the equivalent of menus and palettes in WIMP interfaces but are persistent, i.e. they remain visible outside the interaction area, which has a huge impact on usability for individuals and teams. We do not analyse these individual usability advantages here but consider how paper influences orchestration, i.e. from the teacher perspective.

The tangible interface enabled students to explore rapidly many warehouse designs, but tinkering is not learning. Learning required reflective activities that have to be enforced by the teacher. One student in each team (usually four teams per classroom) had to copy the warehouse layout and performance values by passing a pen on the beamed information, to bring this sheet to the blackboard and to copy the information again on the board (Fig.2 left). The teacher then asks the students to compare their results and to explain why a particular design was better than another (Fig. 2 right). While a client-server architecture would display the same data faster, this media discontinuity affords richer forms of classroom organisation: *handling paper makes the workflow visible and tangible.*

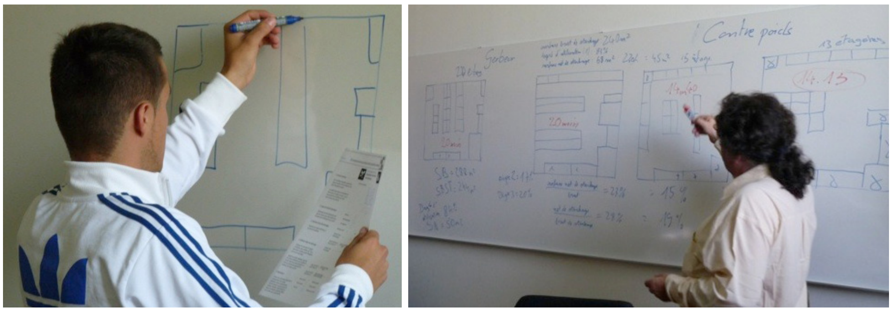

Figure 2. Each team reports their results by using annotated sheets. Comparing the four solutions, the teacher pushes students to discover the properties of an efficient warehouse.

After a few years of collaboration with teachers, the curriculum has been re-structured on Tinker sheets (Fig. 3). Concretely, this curriculum has the form of an A4 binder. To run an activity defined in the curriculum, the teacher selects a sheet and places it partly under the lamp. The part that is viewed by the camera contains all the information that the system needs in order to run the learning scenario associated to that page. After the lesson, the teacher may annotate this sheet with comments for the next year, make copies, etc. Paper makes tangible the educational design cycle: prepare the lesson – teach – reflect.

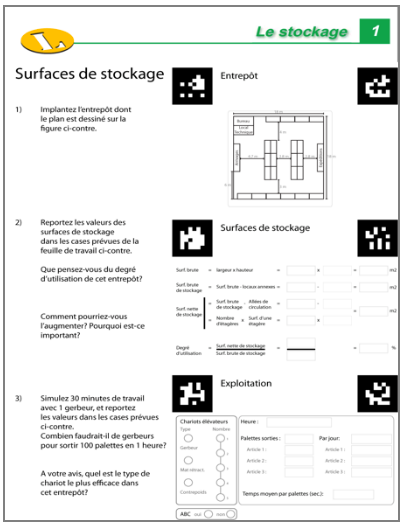

Figure 3. Curriculum Sheets: the Right Part (with Visual Markers) is Input/Output.

These examples revealed an interesting feature of paper-based computing: paper sheets *make the educational workflow tangible*, which implies that this workflow becomes visible and modifiable. Distributing sheets, collecting them, storing them or annotating them are common practices in school ecosystems. Contrarily to the myth of the paperless office, we argue that paper-based interfaces are well suited for *routine worlds*, i.e. environments where the main activities are recurrent and where users develop solid habits.

Circles of Usability

The relevance of paper-based computing for orchestration has only been presented here as an example. We aim to abstract and generalize these findings to a broader range of CSCL technology. We therefore define *three circles of usability*. The first circle concerns the understanding of how individuals interact with (learning) environments. For instance, the tangible shelves of the TinkerLamp enable faster manipulations than a multi-touch table, they provide a real 3D perspective and they off-load students from tackling the scale issues they face when drawing warehouses. The second circle concerns team processes: how do collaboration tools shape interactions among learners; CSCL became paraphrased as *design for conversation* (Roschelle's,1992). The third circle concerns the integration of CSCL environments in the classroom or *design for orchestration*.

What is considered as '*the user'* is an individual person at Circle 1, a team at Circle 2 and the classroom at Circle 3. Referring to *the classroom as the user* means that we aim to understand its processes and constraints. At Circle 1, the constraints were the individual's cognitive load, background knowledge, experience, motivation, etc. At Circle 2, the constraints were related to the team's need to build enough shared understanding to carry the task at hand, the peers' level of interdependence, etc. At Circle 3, teachers have to cope with many constraints: curriculum relevance, time budget, time segmentation, physical space, discipline, security, etc. Understanding the relationship between CSCL design and the management of these constraints is what we refer to as *usability at the classroom level*.

To understand the design features that make a tool 'usable' at the classroom level, we use three concepts: orchestration, awareness and workflows. Before describing them, we need to clarify two points. First, there also exists several circles around the classroom (the school, the community, the society, etc.) that we do no describe here: they receive a lot of attention in CSCL while the classroom circle has been neglected. Second, the term "classroom" is used as a flag: it does not exclude activities outside the classroom (field trips, museum visits, homework, etc.) or corporate training (workshops, seminars, etc.). However, our analysis is restricted to situations where a person (teacher, teaching assistant, parent, workplace supervisor, etc.) has the *responsibility* to bring other persons to reach learning goals (Hoppe, personal communication). For the sake of simplicity, we refer to this responsible person as the teacher. This paper is hence not addressing informal learning.

Orchestration

What does a teacher do if he conducts a CSCL script designed for teams of three, with three roles per team, when suddenly one student drops out the class? What if the next script activity is 40 minutes long but the class ends in 30 minutes? What if two students refuse to work together? Classroom *orchestration* refers to the real time management by a teacher of multiple learning activities within a multi-constrained environment. Classroom management is as old as schools, but it became salient in CSCL when scenarios (or scripts) began integrating individual (e.g. reading), collaborative and class activities (e.g. readings, lectures). This integration requires adapting the script on the fly in order to cope with many constraints. We enumerated many of them (Dillenbourg & Jermann, 2010):

- *Curriculum constraints*: how relevant is the topic with respect to the learning objectives listed in the curriculum? Do students have the prerequisites? Etc.
- *Assessment constraints*: are my learning activities compatible with exams? Does my CSCL tool require a reasonable workload? Etc.
- *Time constraints*: how much time is necessary? How much time is left before the break and how much flexibility do we have around these two factors? How much time is lost simply to install the tool? Etc.
- *Sustainability constraints*: how much time and energy must teachers engage to prepare and run this method? How long can they do it? How much does it cost? Etc.
- *Space constraints*: do I have the space necessary in my classroom do set up these activities? Can I move furniture? Can I walk around the classroom? Is there enough daylight? Etc.
- *Discipline constraints*: Can I keep control of my class? Is the level of noise in the classroom below what is tolerated by the school director? Etc.

These constraints could be considered as practical problems, poorly related to learning theories. It is true that they typically correspond to factors that we treated as "controlled variables" when conducting field studies. However, ignoring these factors probably explains why CSCL has difficulties in scaling up from field experiments to broader educational impact: they not explain why students learn, they may explain why a method could fail. This paper illustrates how CSCL could pay more attention to design features that allow teachers to manage multiple classroom constraints.

Awareness Tools: Less Ambitious than Student Modeling

Orchestration could be described as a regulation loop: the teacher monitors the classroom, compares its state to some desirable state in the scenario, and adapts the scenario accordingly. This loop defines two points of orchestration: *state awareness* and *workflow manipulation*.

Individualizing a pedagogical scenario relies on student modeling: the system (or the teacher) aims at inferring from the student's behavior what (s)he has understood and not understood. This in-depth understanding of students is not possible when orchestrating classroom activities with 25 students. The analysis of traces in CSCL tools has progressively lost depth and gained breadth, for instance proposing visualizations of conversation patterns without analyzing the semantics of utterances (Bachour et al, 2010). This evolution brings us closer to the CSCW notion of "awareness tools" (Greenberg, Gutwin & Cockburn,1996): informing users about the activity of their co-workers: who is on-line, on which document or paragraph are my peers working on, are they available, etc.? Awareness is less ambitious than student modeling since it shares behavioral information among users without cognitive diagnosis. The information overload that awareness may trigger has been tackled through the concept of 'filtering', i.e. selecting the relevant information to share. By downgrading 'student modeling' to 'awareness', we stress the need for *minimalism* in the design of orchestration technologies.

Educational Workflows: The Light and Dark Sides of Integration

Let us illustrate workflows in the context of *"integrated learning*" scenarios, i.e. CSCL scripts that combine team learning with individual activities and class-wide activities. One example is ArgueGraph, which scaffolds argumentation by forming pairs of students with conflicting opinions. This script includes the following phases: (1) students individually answer a questionnaire; (2) the system plots them on a 2D opinion map based on their answers (the teacher having previously defined an X and a Y value for each answer); (3) the system forms pairs of students based on their distance on the graph; (4) pairs of students answer the same questionnaire again; (5) the teachers conducts a lecture based on all answers collected by the system during phases 1 and 4. The individual, group and class-wide activities are computationally "integrated" because the data produced in an activity are necessary inputs for another activity. For instance, in ArgueGraph, Phase 1 answers are used by the system to build the map in Phase 2 and to make teams in Phase 3 as well as for the debriefing (Phase 5). The notion of workflow – another CSCW concept – fits well with CSCL scripts because it does not only refer to a sequence of activities, but also to the underlying flow of data across these activities.

Since they run in the 'back office' of scripts, workflows could be seen as a technical rather than a pedagogical issue. This is not the case: workflows constitute both the strengths and weaknesses of CSCL scripts. Regarding the strengths, a workflow is the condition for integrating heterogeneous activities in a consistent whole. As for weaknesses, workflows usually are internal to the software (not accessible from outside except as log files) and 'hard wired', i.e. difficult to modify. In other words, workflows both enable the execution of CSCL scripts and reduce their flexibility, which creates new constraints to teachers.

A few years ago, we conducted the ArgueGraph script with paper instead of computers and it worked surprisingly well. We distributed the questionnaires as paper sheets. When all students had completed it, we gave them the scheme for scoring their answers in the same way the ArgueGraph software did. They communicated their graph position to the teacher who plotted them on the blackboard and formed pairs. The extra work for manually counting answers and forming pairs was compensated by the ease of manipulation. Nonetheless, we lost a key functionality, the collection of students' justifications for the final debriefing. To avoid this, recent developments in 'paper computing' (next section) combine the advantages of an executable script with the advantages of manipulating real paper.

Paper Computing: Making Educational Workflows Tangible

Tinker illustrates that paper implements tangible and visible workflows: as explained in the first section, paper sheets pass from one activity to the next one; they get annotated, distributed, shared, etc. An additional example concerns the classroom-homework-classroom workflow. In Figure 4, the apprentices save the warehouse layouts they have designed and select what they consider as their four best layouts. The system generates an "individual fieldwork sheet" that the apprentices print and take away: they have to compare these saved layouts to the warehouse in which they work in order to connect school knowledge with experience. Since this printed sheet has the same tags as the other TinkerSheets, they can be used as input for the next school activity.

A second orchestration example addresses a well-known problem in learning from simulations: students can run a simulation many times without much reflection (De Jong & van Jooligen, 1998). As a tangible interface made our simulation especially playful, this risk is high in Tinker. We therefore developed the *paper orchestration keys (POKs).* The "simulate" POK is used to force hypotheses*:* teams cannot run the simulation without showing this POK to the camera. The standard scenario is that the teacher has the key in hands when walking from one team to another and that apprentices hence have to call him when they want to run the simulation. Before giving the key, the teacher will for instance ask them to predict if the warehouse performance (average time to move a box from the shelves to the truck) will be higher or lower than in the previous run, and to explain why. The key empowers the teacher in his management of teams and makes the scenario easy to modify: the teacher may decide to leave a copy of the key to a good team, to give a key to all teams, to take back a key, etc. This could be achieved with options in the software interface but the paper key makes these workflow changes visible for all actors. The teacher and all other teams see at any time who has the key and who does not; they can take it or give it much faster than by tuning options in sub-menus of an application. Note that this POK empower teachers in orchestrating constructivist activities, not in lecturing.

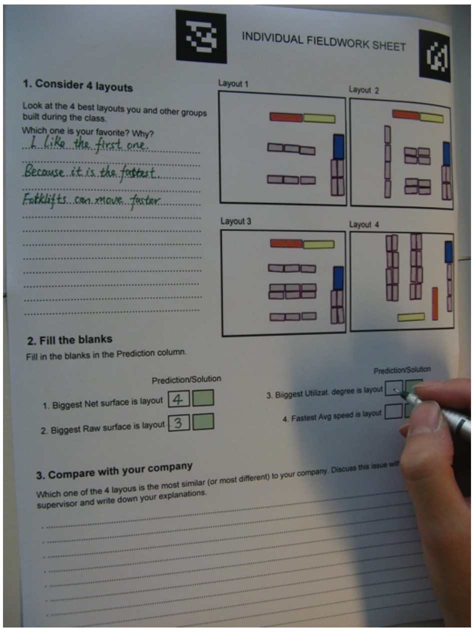

Figure 4. The Classroom Activity Generated Homework Sheets That Will Be Reused by the Environment.

POK s have also implemented in an augmented reality CSCL environment (Figure 5, left) that uses paper to teach geometry in elementary schools (properties of triangles/quadrilaterals, surfaces, angles, symmetry axes,…). The research question is concerned with the effectiveness of learning activities that use paper sheets as tangible objects: paper-made polygons can be rotated, folded (axes), cut, etc. Other types of paper sheets are used as operators. For instance, the students could show a card to overlay a grid over a paper triangle in order to estimate its surface by counting squares. While geometrical objects are made of simple paper, operations are printed on POKs similar to collectable playing cards. Teachers use these POKs to orchestrate the activity in different ways: they may show a POK to the system to display the length of each segment (providing feedback to the kids); they may decide to provide students with quantitative POKs (e.g. measuring the surface) only after they qualitatively understood the notion; they may distribute different POKs to different members of a team to define roles, etc. POKs have to be more robust than sheets representing polygons because they are used for longer periods (paper polygons are not reused since they are cut, folded, colored, etc. by the students) and more rigid to be as easily manipulated as play cards.

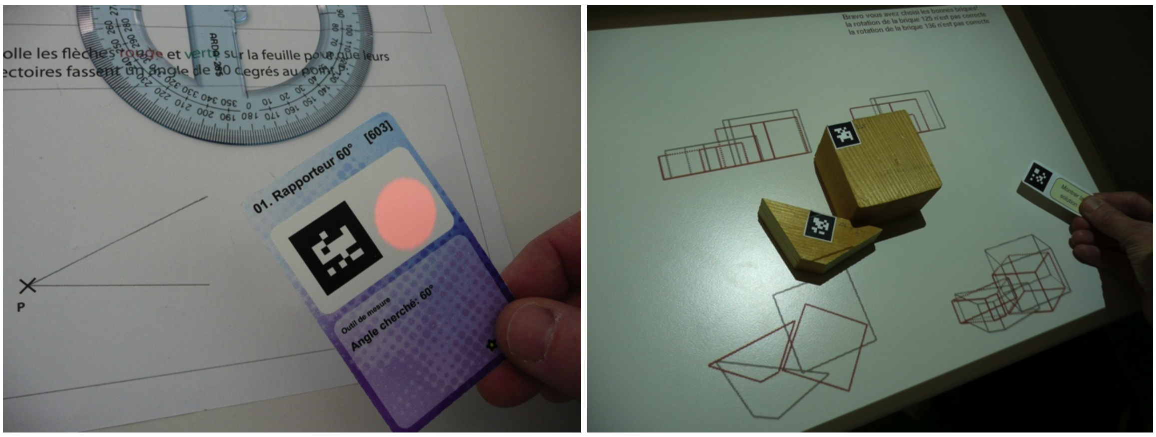

Figure 5. Left: this card asks for feedback on the angle to be constructed (the beam red dot indicates it is not correct). Right: this card provides scaffolds (as explained below).

The same approach is used in another environment: for training apprentices in carpentry (Figure 5, right). The research question here relates to the development of spatial reasoning skills with augmented drawings. The apprentices manipulate wooden blocks and the computer displays their projections in the three orthogonal planes (what they have to draw at school). An example of a script is that a team "saves" a construction and gives it to another team that has to assemble the blocks in a way that matched first team's projections. The POKs presented by the teacher displays (in red) a scaffold for the second team: the difference between their current construction and the one they have to produce.

These examples illustrate that paper interfaces make the workflow tangible and a tangible workflow is visible to all actors and easier to modify. Paper-based interfaces are promising tools to combine this tangibility / visibility without abandoning computational power. Of course, we do not pretend that paper intrinsically facilitates orchestration; it is a matter of design. For designing interfaces, we propose a simple model (PAW). Interactive paper sheets are covered by three layers of information: those printed in advance (P), those beamed by the augmented reality environment (A) and those written or drawn by the learners and/or the teacher (W). Designing paper-based interfaces is about understanding the complementarily of the 3 layers. The P layer contains hard elements from the script, the A layer makes the script interactive but A info is lost when kids leave the system while the W layer will remain after the session The need to produce tangible traces of learning activities (e.g. for parents) typically is a classroom constraint that CSCL did not pay much attention to.

Modest Computing: Minimalism in Ambient Awareness

So far, we stressed the usability of paper for orchestrating activities. To broaden our argument, we now present a very different orchestration tool. It originates from our observations of teamwork at the university level. Typically a first year course in physics is composed of two hours of lecture plus two hours of exercises per week. During these exercises, most students work in small groups (two to four students) on a list of 8-10 exercises. When students are stuck, they raise their hands and one of the teaching assistants (TAs) comes when (s)he is available. In terms of orchestration, this is fairly simple compared to complex CSCL scripts, yet it is far from being optimal. Twelve recitations sections have been videotaped from three different courses involving around eighty students altogether (Alavi et al, 2009). While waiting for the TA, students spend 62% of their time visually chasing the TA because, if they do not grab him or her as soon as (s)he is available, (s)he might go to another team. Other problems were observed such as unanswered questions (students give up) or the TA helping a team that has been waiting much less than another one.

Alavi designed two tools to address these problems. The first one, named Lantern (Fig. 6 left) is a small device (in size of 0.5 L bottle) consisting of five LEDs installed on a stub-shape PCB and covered by a blurry plastic cylinder with one microprocessor to control the LEDs. By turning the cover, students indicate which exercise they are working on: every colour corresponds to one exercise. The height of the colour bar indicates how much time that has been spent on the current exercise. When a team wants to call the TA, it presses the Lantern which starts blinking. The blinking rate increases slowly indicating the waiting time. The second tool, named Shelf (Fig. 6 right) uses exactly the same visual codes as Lantern, but students communicate with a clicker and the status of teams is displayed centrally on a display. We provided both tools to two courses of physics. In both classes, students and TAs used Shelf for three weeks, after that they switched to use Lantern for four weeks. In total, Shelf has been used for around 12 hours and Lantern for 14 hours. The main result is that the estimated time wasted in chasing the TA was reduced from 62% in our early observations to 16% in the Shelve condition and to 6% in the Lantern condition. Students simply continue to work while waiting.

Figure 6. The Lantern Device (Left) and the Shelve Environment (Right).

When orchestrating 10 teams of students, the TA iteratively faces two questions: which team should I help now and what should I tell them. Shelf and Lantern are only concerned with the first question, which is easier than the second one. Moreover, these tools do not decide where the TA should go next. They more modestly provide TAs with some "awareness," as defined earlier, of the teams' behaviour. They are not smart tools; they neither interpret activities nor predict the need to intervene, but they *simply make visible things that would otherwise remain invisible*: working time and waiting time. The decision remains in the TA's hands. Our *minimalism* does not only apply to the functionality of Lantern but also to its design. *We deliberately reduced the resolution of the display:* instead of displaying the precise exercise number and the exact waiting time, Lantern provides degraded information. The term "ambient" is used for displays that do not monopolize the visual attention of users. Given the main effects of the Lantern, we are even tempted to believe that this minimalism is a condition for orchestration, but this is only a hypothesis at this point.

The second interesting result in terms of orchestration is that the physical layout had an impact on the social processes. Shelf induced some competition between teams, while Lantern triggered collaboration between teams: when Team could see that a neighbouring Team was moving to a next exercise (they changed colour), Team 1 would sometimes ask Team 2 for a suggestion. Lantern generated a social/spatial organisation of the classroom into spatial clusters of two to three teams. The fact that t two tools that provide almost the same information generate different social processes illustrates the physicality of orchestration: it is about mobility, gaze directions and distances between all classroom actors.

Let us analyze more deeply the fact that teams peripherally perceive their friends, hence see when a neighboring team moves to the next exercise, which triggers inter-team interactions. Similarly, in the Tinker studies, students did also look over the shoulders of other students to see the warehouse being built by other teams. Unlike desktops, tabletop environments induce indeed two interaction spheres: a first sphere of users who manipulate objects on the table and a second sphere of students who can see or who can hear what is done in the first sphere. "*Looking over the shoulder*" had a positive effect in the Lantern study but could have a negative effect in the Tinker classrooms (students copying the warehouse of others instead of exploring). Whether they are deliberate or accidental, "looking over the shoulders" and "over-hearing" often happen in classrooms. They are realities of orchestration, not investigated in CSCL, that illustrate well the third circle of usability. Let us review the three circles in terms of what users visually perceived. At Circle 1 of usability, HCI is concerned by how well the user perceives the display (readability, understanding of symbols, etc.). At Circle 2, CSCL/CSCW investigated if team members should or not perceive the same things (WYSIWIS: "what you see is what I see"). At Circle 3, a new concern is to analyze when team members look at the display of another team. The same circles can be defined for audio perception, "overhearing" being at Circle 3.

Figure 7. Three Designs of the TinkerLamp: the Used Model (Left) and Two New Models.

Investigating "orchestration" requires an analysis of how CSCL designs influence "looking over the shoulder" and "overhearing", namely how they modify the line of sight for students and, very importantly, for the teacher. Let us illustrate this with the design of the Tinker Lamp. In the reported experiments, the teacher placed four lamps (Fig. 7 left) in the classroom. There were several problems in putting the beamer above the surface. Therefore, our new designs include the beamer on the table and a mirror above the table (we do not project from below the table because we have to display information on the shelves and on paper sheets). The two new lamps are illustrated in Fig. 7 middle and left. Their designs induce different orchestration processes. The black model prevents the teacher from seeing in a glance what students are doing while the white model does not break the line of sight. The white model is better suited for the scenario we used in logistics training where teams simultaneously use up to five Tinker in the classroom. Orchestrating a classroom with five black lamps would require the teacher to run around the class for monitoring what students do. Conversely, elementary classrooms have a corner with a bookshelf, a sofa and a table with a computer. The black model is better suited these classrooms: two students can work there without perturbing too much the rest of the class.

Conclusions

For several years, CSCL has been striving for a better integration of our tools into educational ecosystems: CSCL does not appear in a vacuum but is part of an ecosystem and should be integrated into other activities (individual and class-wide), with or without computers, inside or outside the classroom, etc. This evolution reflects the maturity of CSCL (team learning is not the unique approach) as well as the technological evolution (mobile devices, tabletops, etc.). This integration requires deepening our understanding of orchestration.

This paper addressed the question "how do CSCL tools influence orchestration?" from a perspective that may be shocking: we stressed the practical aspects of activities rather than the learning processes themselves. But our experience is that the success of CSCL tools is hidden in these implementation details. Ignoring them systematically leads to develop environments that increase the teacher's "global orchestration load." Every menu to pull down and option to select increases the workload of a teacher who acts live in front of 20 or 30 students. The experiments we conducted led to *simple design principles*: (1) strive for *minimalism* in design (few functionalities, reduced resolution of information), (2) *care for visibility* by taking care of lines of sight and implement *reification* (make visible aspects that are usually invisible) (3) make the educational *workflow tangible* (4) *empower teachers.* The last point is the consequence of the previous ones. Empowering teachers is neither wishful thinking, nor pushing authoritative visions of education. What we mean is that, in CSCL environments, the teacher should literally hold the scenario in his or her hands, such that it is easy to manipulate, and not simply have a few options to select in a predetermined script before unfolding him or her. The evolution of computer science provides us with new tools to embed these principles in CSCL environments: paper-based computing, tangibles, ambient displays, tabletops.

These "*design for orchestration*" principles do not form a theory at this stage. Nonetheless, as one step in that direction, we proposed the *third circle of usability*, i.e. a set of concepts that are part of orchestration and that have not been considered so far at lower circles of orchestration. By using somehow provocatively the word 'usability', we do not discard that classroom orchestration raises other pedagogical factors, but we stress that fact that this classroom usability is a necessary condition to implement effective learning scenarios.

References

Alavi, H. S., Kaplan , F. & Dillenbourg, P. (2009) Distributed Awareness for Class Orchestration. In U. Cress, V. Dimitrova & M. Specht (Eds). Proceedings of the 4th Europen Conference on Technology Enhanced Learning, pp. 211-225, EC-TEL 2009, Nice, France, Sept. 29th – Oct. 2nd. Springer, LNCS 5794

Bachour, K., Kaplan, F., Dillenbourg, P. (2010) "An Interactive Table for Supporting Participation Balance in Face-to-Face Collaborative Learning," *IEEE Transactions on Learning Technologies*, 3 (3). 203-213

De Jong and van Jooligen (1998). Scientific Discovery Learning with Computer Simulations of Conceptual Domains. *Review of educational research*, 68, pp. 179-202.

Dillenbourg, P. & Fischer, F. (2007). Basics of Computer-Supported Collaborative Learning. *Zeitschrift für Berufs- und Wirtschaftspädagogik.* 21, pp. 111-130.

Dillenbourg, P. & Jermann, P. (2010) Technology for Classroom Orchestration, in: M. S. Khine & I. M. Saleh (Eds) *New Science of Learning: Cognition, Computers and Collaboration in Education* (pp. 525 – 552). Springer. Dordrecht

Greenberg, S., Gutwin, C. & Cockburn, A. (1996) Awareness Through Fisheye Views in Relaxed-WYSIWIS Groupware. Proc. of *Graphics Interface* (pp. 28-38), Toronto, CA, May 21-24. Morgan Kauffman.

Lucchi, A., Jermann, P., Zufferey, G., and Dillenbourg, P. (2010). An empirical evaluation of touch and tangible interfaces for tabletop displays. In *Proc.s of the 4th int. conf. on Tangible, Embedded, and Embodied interaction* (Cambridge, MA, USA, January 24 - 27, 2010). TEI '10. ACM, New York, NY, 177-184.

Nussbaum, M. C. Alcoholado, A. Tagle, F. Gomez, F. Denardin, H. Susaeta, M. Villalta, K. Toyama (2010), One Mouse per Child: Interpersonal Computer for Personal Formative Assessment, Submitted

Roschelle, J. & Pea, R. (2002) A walk on the WILD side: How wireless handhelds may change computer-supported collaborative learning. *International Journal of Cognition and Technology*, 1(2) pp.145-168.

Roschelle, J. (1992) Learning by Collaborating: Convergent Conceptual Change. *J. of the Learning Sciences*, 2, 235-276.

Zufferey, G., Jermann, P., Lucchi, A., and Dillenbourg, P. (2009) TinkerSheets: using paper forms to control and visualize tangible simulations. In *Proceedings of the 3rd international Conference on Tangible and Embedded interaction*(Cambridge, UK, Feb, 16 - 18, 2009). TEI '09. ACM, New York, NY, 377-384.

Acknowledgments

Tinker is developed within the leading house "Technologies for vocational education" supported by the Swiss Ministry of Economical Affairs. Lantern has been funded by the Swiss NSF under the ProDoc PDFM1-118708. Thanks to O. Guédat and W. Hokenmaier who built the hardware as well as to K. Bachour, J. Kurzo, A Ryser, P. Ogay and the teachers who co-designed these tools. Thanks to C. Sanchez, M. Chablais, S. Testuz,