Trace Theory, Coordination Games, and GroupScribbles

Charles M. Patton, Ph.D. **Center for Technology in Learning** **SRI International** **333 Ravenswood Ave.** **Menlo Park, CA 94025** **charles.patton@sri.com** **541-729-1627 (office)** **(650) 859-4605 (fax)**	**Deborah Tatar, Ph.D.** **Center for Human-Computer Interaction** **508 McBryde Hall** **Virginia Tech** **Blacksburg, VA 24060** **tatar@vt.edu** **(540)231-8457 (office)** **(540)231-92186075 (fax)**	**Yannis Dimitriadis, Ph.D.** **GSIC / EMIC group** **University of Valladolid, Spain** **yannis@yllera.tel.uva.es**

I. Coordination in Learning

The role of sociality in learning is a tenet of modern pedagogical theories based in the learning sciences. Collaborative learning is an important area of research that seeks to take advantage of this. Techniques such as the jigsaw classroom and reciprocal teaching emphasize the cooperation and interdependence between learners. But as collaborative patterns of engagement (DiGiano, et al., 2003) multiply and automated tools, such as pattern-based editors (Hernandez, et al., 2006), increase the potential for variation and creativity in the design of collaborative learning experiences, there is an increasing need to better understand and account for the low level coordinative conditions that make them possible. The nature of coordination, especially participant-driven coordination, in collaborative learning, is under-theorized and under-explored.

That is, a collaboration designer can specify the desired overall pattern ("jigsaw", "peer instruction", "literature circle", etc.) at the high-level plan view perspective, leaving the dynamic particulars as an unexamined black box that is either too complex, or too unimportant, or both, to bother with as a design task. There are, however, a number of compelling reasons for attempting to consider the patterns of interaction at a concurrent, dynamical level. First, as an instance of patterned behaviors emerging from simple rules, it is fascinating and important science in its own right (Wolfram, 2002) Second, there is accumulating evidence (Kollar, et al. 2003; Zurita, et al., 2003) that the real power of collaborative learning comes not from the seamless flow implicit in the plan view, but rather from the "seams" in the group understanding that emerge and are closed in the collaborative process. Finally, allocation (implicitly or explicitly) of power and responsibility in classroom collaborative processes is well known to be an important factor in the opportunity to learn (Cohen and Lotan, 1996; Johnson & Johnson, 1999; Slavin, 1996). These processes are essentially hidden at the plan level of collaboration, but may be more amenable to designed exploration with the coordination games that will be suggested here.

In this chapter, we address the problem of how to understand complex, fine-grained coordination by first introducing GroupScribbles, a tool for group collaboration and coordination. We then introduce a formal language, trace theory, for describing the coordinative properties of the interaction. Finally, we present two examples of alternative coordination structures for participant-driven versions of the jigsaw pattern.

II. Group Scribbles

The GroupScribbles[1] (Brecht et al., in press) system was designed to enable collaborative improvement of ideas based upon both individual effort and social sharing of notes in graphical and textual form ("scribbles"). GroupScribbles provides a way for educators to rapidly design new collaborative and group learning activities without the need for additional programming.

All participants in a GroupScribbles session have their own computers. Each computer has a two-paned window. The top is a public work area that is shared between participants and identical on each person's screen. The lower pane is the user's personal work area, or "private board", with a virtual pad of fresh "scribble sheets" on which the user can draw or type. A scribble can be shared by dragging and dropping it on the public board in the upper pane. Other participating clients monitor the space for such activity and update the client's display. Users may interact with public scribbles in a variety of ways, such as browsing their content, repositioning them, or moving one from the public board into their private space. New public boards can be created to support multiple activities or spaces for small groups to work.

[Insert Figure 1 around here]

[1] See, e.g., http://groupscribbles.sri.com

GroupScribbles has been used in many demonstration sessions, informal workshop meetings, and even in real classes, all over the world (Taiwan, US, Spain) since its first release in mid 2006. It is a general-purpose representational tool that can easily be used to express views, use diagrams, point out joint conclusions, try to reach an agreement, obtain common conclusions, or even vote. Three examples demonstrate its functionality and representational flexibility:

Example 1 A World Map. Figure 1 shows a typical warm-up use. A drawn world map is applied as a background image for the public board and participants are asked to write their name on a scribble sheet and then place it on the board near their home location.

Example 2 A Changing Assessment. Figure 2 is drawn from a real classroom use of Group Scribbles in Spain in November 2006. The teacher asked students to assess themselves by posting scribbles along the number line that he had drawn. The left pane shows their assessments prior to the class activity and the right pane after the class activity.

[Insert figure 2 around here]

Example 3 A Planned Activity. Figure 3 shows a more complex activity in which a set of boards and background images was offered to students participants. A central public board provided a general overview of the activities. Students followed the general plan. They also employed stickers for awareness purposes, thus contributing to a better coordination. Lastly, low-level interactions were handled through social protocols, as e.g. the group formation, or the voting procedure of the most important conclusions (Dimitriadis et al., 2007).

[Insert figure 3 around here]

III. Group Scribbles & Coordination: Key aspects of design enable a focus on coordination

The simple appearance of the GroupScribbles application belies the fact that its features were inspired by the goal of enabling theory-based exploration of coordination patterns and their interaction with content.

Computational neutrality. GroupScribbles is an example of *Zensign*, the idea that what you leave out of an interface is as important as what is put in (Harrison, Tatar and Sengers, in press; DiGiano et al., 2007). The encapsulation of free-form content in movable blocks (scribbles) allows for a full spectrum of activities ranging from almost pure coordination (in which the scribbles are used primarily as counters or tokens) to almost pure content aggregation. The machine cannot compute on the contents of the scribble, leading to content neutrality. It is not directly related to a domain, inquiry process or even a particular problem solving approach. For example, instead of embedding a typical set of steps of scientific inquiry in physics as WISE (Linn, in press), GroupScribbles allows teachers and students to set all necessary conditions.

*Small (but generative) set of primitive actions***.** Similarly, the central *put*, *read*, and *take* metaphor, inherited from the underlying tuplespace implementation (Wyckoff, McLaughry, Lehman, & Ford, 1998; Carriero & Gelertner, 1990), together with the shared special partitioning of the public space enables the support of many complex coordination schemes, as shown in the extensive literature on tuplespace-based coordination for computer processes.

Background-structured groupings. Background images for the public space, onto which scribble sheets are placed, can be, and usually are, used to provide location-based metadata for the scribbles placed thereon. This “putting in / taking from a particular place” provides the next level extension of the set of primitive coordination actions. These background images can also point out to a “backdrop” metaphor, because of the way they seem to help contextualize learning activities in a manner similar to how painted scenery helps situate a theatrical performance.

Small footprint. The GroupScribble client software was designed to have a small code footprint, and be usable with a quite modest allocation of screen real estate. As such, it can be employed unobtrusively in conjunction with other applications or even as the coordination component of primarily non-computer-based activities.

Socially-mediated Protocols. What might be considered a design deficit – namely the absence of technology-based mechanism to *enforce* coordination protocols beyond the contention resolution embedded in the primitive actions – is, in fact, a design decision.

Such design decisions position GroupScribbles almost at the opposite extreme of the technology-based coordination protocols that are often employed in CSCW or workflow environments. The analysis of this design tension (Tatar, in press) has been partly studied in a real learning setting, showing an extraordinary potential for fruitful coordination interactions (Dimitriadis et al, 2007) as in example 3 above.

Three considerations came into play in this decision. First, we are most interested in exploring *participant driven* coordination rather than centrally administered coordination. Second, we are interested in exploring *spontaneously generated elaborations* or refinements of coordination protocols. Finally, an important parameter distinguishing alternative embodiments of coordination patterns is the *assignment of responsibility,* e.g. who (or what) is in charge of which aspect of the protocol, a question made moot by assigning enforcement to the technology.

Having such an unobtrusive, supportive, and flexible shared environment on which the participants can play out the full spectrum from content-rich to coordination-centric "games", one final thing is needed to enable creating and analyzing the fine-grained coordination in collaborative learning scenarios: a powerful and broadly applicable formalism – Trace Theory. With this formalism we can begin to investigate a) how control of the pattern might be distributed and b) explore the potential consequences of alternative detailed patterns, all while preserving the overall structure as an emergent property.

IV. Using Trace Theory to describe and specify coordination structures in Group Scribbles

From a formal perspective, coordination games may be described as a set of allowable event sequences, together with distribution of responsibility among the participants for initiating and concluding events, and rules for each participant regarding allowable initiations and conclusions under their control.

Trace Theory (Dill, 1989; Benko, 1993; Benko and Ebergen, 2002) is a formalism that was devised and refined by the integrated circuit design community as a means of specifying, designing, and verifying the design of the collaborative, asynchronous, and delay-insensitive behaviors of interconnected arrays of circuit components. Though we will borrow liberally in simple examples from trace theory concepts and notations, the formalism, and associated tool set is capable of handling patterns with thousands of participants and equally complex rule sets.

The components of the formalism include (1) regular expressions over an alphabet, (2) projection onto a sub-alphabet, and (3) weaving of specifications to yield coordinated sequences. Recall that regular expressions constitute a simple (but powerful) way of specifying sets of sequences (a "grammar") over some alphabet, S, here interpreted as a set of events. The following then constructively define regular expressions:

- The empty sequence, ε, is a regular expression.
- For any a in S, the singleton sequence "a" is a regular expression.
- At this point, if the alphabet consists of the events $\{a,b\}$, the only regular expressions are $[\varepsilon]$ => {""}; $[a]$ => {"a"}; $[b]$ => {"b"}.

- If *u* and *v* are regular expressions, then the concatenation *u;v* is also a regular exp. If U and V are the sets of sequences corresponding to *u* and *v*, the set corresponding to *u;v* is {*xy* with *x* in U and *y* in V}.
- If *u* and *v* are regular expressions, then the alternation *u*|*v* is also a regular expression. If U and V are the sets of sequences corresponding to *u* and *v*, the set corresponding to *u*|*v* is {*x* with *x* in U or *x* in V}.
- If *u* is a regular expression, the Kleene Closure, *u** is a also a regular expression. If U is the set of sequences corresponding to *u*, the set corresponding to *u** is the set of zero or more concatenations of elements of U (and so contains the empty sequence.)

Regular expressions are good building blocks for specifying coordination patterns. In particular, they are concise and comprehensible (or at least familiar) and they easily suggest many patterns of interest:

- "a and b take turns" => [*a;b*]*
- "a is the card dealer, b and c are players => [*a;b;a;c;a;*[*b;a*]*;*o;a*;[*c;a*]*;*o*]*

They are, however, inherently sequential and don't immediately provide a descriptive emphasis on the distribution of responsibility. For this, we need more machinery.

Projection onto a sub-alphabet is a second component of the trace theory formalism. If *U* is a set of sequences over an alphabet S and T is a subset of S then the projection of *U* onto T is the set of sequences *U*↓T resulting from removing all elements that are not in T from each sequence in *U*.

Projection is a concept useful in a number of ways in specifications. First, projection is often useful in designing an implementation of a specification to include "auxiliary" events that aid in the implementation but don't otherwise affect the overall pattern. (This will be illustrated in the jigsaw example, below.) To capture this latter constraint (the lack of affect on the overall pattern), we can insist that the projection of the set of extended sequences onto the original alphabet is the same as the original set, thus ensuring the original behavior. Second, projection will be useful to combine validated implementations of simpler components to build up implementations of more complex patterns. For example, having settled on an implementation of the two-by-two jigsaw (below), we consider how we might link together four of these to produce the equivalent of a four-by-four jigsaw. To do this, we would need to break open some of the linkages so that, say, *D!* is connected with a corresponding *D?* in another block. To specify and validate the block to block pattern, it is again useful to project away from the alphabet of events purely internal to a block. Note that this is essentially the contrapositive of the first use in that rather than extend a pattern with auxiliary events and then check that the desired pattern is unchanged, we combine already extant implementations and check that they produce the desired higher-order pattern. The asynchronous design community has already specified and thought through the implementation of quite a variety of such components, such as sequencers, arbiters, wires, forks, mutual exclusion patterns, etc. (EDIS, 1998), as well as specification and exploration of meeting scheduler problems (Benko and Ebergen, 1994) not unlike that of the coordination problem. Many of these could prove very useful for the building interesting coordination games for collaborative learning.

Finally, projection is useful for elucidating one of the most unique components of trace theory: the weave. If T and R are both sub-alphabets of S, and *U* is a set of sequences over T and *V* is a set of sequences over R, then the weave *U*||*V* is the set of those sequences over S whose projection onto T are in *U*, and whose projection onto R are in *V*. If R=T, then the weave is the set intersection of *U* and *V*. If R and T are disjoint, the weave is the set of all possible interweavings of elements of *U* with elements of *V* (hence the name.)

We can use Trace Theory to describe coordination in Group Scribbles by starting with a base rule for coordinated games. From this point of view, the simplest game is a repeating event, *A*, where the set of sequences is given by [*A*]* (zero or more successive instances of event *A*). To make this into a *coordination* game, we assign responsibility for initiation of *A* (denoted *A!*) to one player and responsibility for conclusion of *A* (denoted *A?*) to another. To mediate the participation of our players (and provide for tracking progress) it is useful to have an assigned, shared location (physical or virtual) into which one participant can put a token or other indicator to signify the initiation of event *A* and from which the other can take said item to signify the conclusion of event *A*.

The base rule implies that only one thing can be put into the location associated with *A*, and only if the location is empty, and nothing can be taken from location *A* unless something has been placed there (this may be recognized as a special case of the Petri Net formalism (Reisig, 1985)). The rules for the two participants can then be described as

- Participant 1: Follow the base rule, and when ready, put an item in the location associated with *A*. Repeat.
- Participant 2: Follow the base rule, and when ready, take an item from the location associated with *A*. Repeat.

We have, in essence, one of the first games we play with babies: to hand them something which they eventually drop and then hand it back to them.

While it is well known (Sipser, 1997) that any set of sequences determined by a regular expression can be decided (and hence produced) by a finite state machine, the descriptive approach described here allows us to distribute responsibility for the production of the sequences across multiple participants (persons and/or machines) and to consider alternatives to that distribution of responsibility.

Weave specification is needed when there is actual coordination between participants. For example, interdependence of action occurs when the adult waits for the baby to take and hold the ball, so that the adult's action of putting is the baby's action of taking. Interdependence does not occur when the baby has dropped the ball and indeed play enters into a new phase when the baby learns to hand or throw the ball back. In the two-person case of "catch," the actions are isomorphic and joined to one another. If more players are involved, then the actions of each are joined only to the actions of the person who puts the ball in one's hand and those who takes it from one's hand. That is, for the three-person case, event [A;B]* is when Participant 1 gets the ball and awaits Participant 2's readiness. When Participant 2 is ready, then event [B;C]* is begun, that is the conjoined offering of the ball by Participant 1 and taking of it by Participant 2 followed by Participant 2 waiting for Participant 3 to be ready. When Participant 3 is ready, [C;A]* is begun, and so forth.

In this description, the system is working properly as long as the experience of each of the participants (the projection onto their own event alphabet of interest) is correct. Specifying coordinated behaviors through a weave of sequential or otherwise well-understood behaviors, when possible, is then a powerful tool for simplifying the design, implementation, and validation process as it focuses attention on the essential coordination aspects of the overall behavior. Figure 4 illustrates two asynchronous moves in the GroupScribbles implementation of the sequencer game, described after Benko and Ebergen (2002), by $[\boldsymbol{R_0;G_0}]^* \parallel [\boldsymbol{R_1;G_1}]^* \parallel [\boldsymbol{S;}(\boldsymbol{G_0} \mid \boldsymbol{G_1})]^*$ with responsibility distributed among five participants.

[Insert Figure 4 around here?]

V. Alternative coordination structures for participant-driven JigSaw

The jigsaw (Aronson et al, 1978; Slavin, 1980) is a collaborative learning activity in which participants are dependent on one another to produce a satisfactory outcome. To provide a simple, concrete example of the opportunities and challenges of specifying coordination patterns dynamically, consider the case of a two-by-two jigsaw. In this case, each of the four participants alternately acts in two roles, as an expert (say as a dissolved oxygen or benthic organism expert in a water quality activity) and as a project participant (surveying water quality at a particular site). At various times, the dissolved oxygen and benthic organism experts, respectively, meet to discuss issues related to their particular focal areas, and then site teams meet to carry out some aspect of the water quality survey of their assigned site, and the process repeats.

If we denote by *D* and *B* meetings of the dissolved oxygen and benthic organism experts, respectively, and by *1* and *2* meetings of the site one and two project teams, the experience of the site one benthic organism expert over time can be described *B;1;B;1;B;1;…* or more succinctly in regular expression notation as [*B;1*]*. Similarly, the experience of the other three participants can be described as [*B;2*]*, [*D;1*]*, and [*D;2*]*.

In the typical case of classroom enactment, the coordination pattern is centrally controlled – the teacher or other facilitator decides when the participants should switch roles. However, quite a bit of coordinative behavior (initiating or stalling, claiming materials or locations, passively tagging along) is likely to occur between the moment when the teacher calls out "time to switch roles" and the time that the actual transition is completed, none of which is either specified or captured in the plan view, and some of which may be very important to learning.

Alternative distributions of responsibility have potentially different affective properties for the participants. We will use the notation, above, to specify the responsibilities for each participant. Thus, the pattern [B!;2?]* assigned to a participant would mean that they are to initiate a benthic organism expert meeting, then conclude a site 2 meeting, and repeat the pattern. In addition to the base rule, we add another rule consistent with the meeting interpretation of the events: if a participant initiates a meeting, they wait until the meeting is concluded to go on to the next step.

A. The lead students' pattern

In this version of the game, the responsibilities are distributed as follows:

- Student 1: [*D!;1!*]*
- Student 2: [*D?;2!*]*
- Student 3: [*B!;1?*]*
- Student 4: [*B?;2?*]*

Figure 5 shows the chronological ordering of two cycles of meetings consistent with playing the coordination game as described above. Note, particularly, that the meetings need not be synchronous (as might be implied from the static, plan view of the pattern). Even so, assuming that each of the meetings actually took place somewhere in the colored band regions (that is to say between the time they were initiated and the time they were concluded) then the *experience* of each of the participants conforms to the intended pattern of interaction.

[Insert figure 5 around here]

B. Equitable leadership pattern

It is important to note that in the lead students' pattern, there are, from the perspective of initiation and conclusion, at least, three classes of participants: those who only initiate, those who only conclude, and those who both initiate and conclude. From the value neutral perspective of the pattern specification, these may be considered equivalent, but in the context of the classroom and the status negotiations that can occur there, they may not be considered equivalent. For example, students who initiate meetings may well be thought of as having higher status than those who don't. We should then ask if there is a more equitable (i.e. only one class of participants) distribution of responsibility that would result in the same overall interaction pattern. Consider the following pattern instead:

- Student 1: [*D!;1?*]*
- Student 2: [*D?;2!*]*
- Student 3: [*B?;1!*]*
- Student 4: [*B!;2?*]*

Inspection reveals that, in this case at least, it is possible to distribute responsibility somewhat more equitably and still have the appropriate overall pattern emerge. We may then ask what set of rules would lead to this more satisfactory distribution of work.

For other overall patterns and/or numbers of participants, distributions of responsibility with high levels of symmetry can lead to unworkable solutions. In the 3-fold dining philosopher's version of meeting scheduling (Benko and Ebergen, 2002), for example, with the interpretation that students are forks and "eating" is a meeting between two students, specifying complete symmetry among the forks leads to the possibility of deadlock wherein all the students initiate a meeting which, as a consequence of the fact that they are then not available to meet otherwise, can never be concluded.

To carry these considerations to the next level clearly requires the two elements we have described: a compact, hierarchical formalism for specifying more complex coordination games at the level of detail illustrated above and flexible, low-burden support technology for playing the games as specified. The formalism needs to be compact so as to allow a designer to think in terms of "chunked" concepts and hierarchical so that once a sub-pattern is validated and understood, it can become another conceptual "chunk" in thinking through patterns with larger scope and more participants. The technology support needs to be flexible both to keep the issue of distribution of responsibility a live topic for as long as possible, and to allow its use in a variety of learning circumstances. It needs to be low-burden both for the participants and for the teacher/facilitator since we have purposely dissociated the coordination pattern from explicit content ("Macbeth" vs. "water quality" plays no essential role in the patterns) and, as such, the coordination game needs to remain in the background to be perceived of as useful.

VI. Summary, conclusions, and future research

In our explorations, the dynamic coordination as experienced through enacting such a specified coordination game is much richer and more nuanced than might be expected from the static view, even while the emergent pattern conforms to the static view. Since the put and take actions (and associated locations) are automatically logged on the server, it is a simple matter to implement a monitoring functionality that checks the emerging event pattern against the specification. Together, trace theory specifications, the canonical interpretation as coordination games, and GroupScribbles as a game board show considerable promise as a means of exploring the detailed dynamical role of coordination in learning.

References

Aronson, E., N. Blaney, C. Stephin, J. Sikes & M. Snapp. The Jigsaw Classroom. (1978). Beverly Hills, CA: Sage Publishing Company.

Benko, I. (1993) The committee problem and delay-insensitive circuits. Master's thesis, Department of Computer Science, University of Waterloo.

Benko, I. and Ebergen, J. (1994). Delay-insensitive solutions to the committee problem. In *Proc. International Symposium on Advanced Research in Asynchronous Circuits and Systems,* pages 228-237

Benko, I., and Ebergen, J. (2002). Composing snippets. In J. Cortadella, A. Yakovlev, and G. Rozenberg, editors, *Concurrency and Hardware Design*, volume 2549 of *Lecture Notes in Computer Science*, pages 1-33. Springer-Verlag:NY.

Brecht, J., DiGiano, C., Patton, C., Tatar, D., Chaudhury, S. R., Roschelle, J. and Davis, K. (in press) Supporting Collaborative Learning Activities with a General-Purpose Interface. To appear in (M. Ally, Ed.) Mobile Learning in Education and Training. Athabasca Canada: Athabasca University Press.

Brecht, J., Pea, R., & Chung, M. (2002). CML – The ClassSync Modeling Language. Presented at the Computer Support for Collaborative Learning (CSCL) conference, Boulder, CO.

Carriero, N., & Gelernter, D. (1990). How to Write Parallel Programs. Cambridge MA: MIT Press.

Cohen, E. G., & Lotan, R. A. (1995). Producing equal-status interaction in the heterogeneous classroom. *American Educational Research Journal*, 32(1):99-120.

DiGiano, C., Yarnall, L., Patton, C., Roschelle, J., Manley, M. (2003). Conceptual tools for planning the wireless classroom. *Journal of Computer Assisted Learning*, 19(1):284-296.

Y. Dimitriadis, J.I. Asensio, D. Hernández, J. Roschelle, J. Brecht, D. Tatar, S. Chaudhury, C. DiGiano, C. Patton (2007), From socially-mediated to technology-mediated coordination: A study of design tensions using Group Scribbles, In xxx (Eds), Proceedings of the International Conference on Computer Support for Collaborative Learning, July 2007, Rutgers, NJ:

Dill, D. (1989). *Trace Theory for Automatic Hierarchical Verification of Speed-Independent Circuits*. ACM Distinguished Dissertations. MIT Press:Boston.

EDIS (1998). Encyclopedia of delay-insensitive systems. Author. Retrieved January, 2006 from http://edis.win.tue.nl/edis.html

D. Hernández, E.D. Villasclaras, I.M. Jorrín, J.I. Asensio, Y. Dimitriadis, I. Ruiz, B. Rubia (2006), COLLAGE, a collaborative learning design editor based on patterns. *Educational Technology and Society*, 9(1): 58-71

Johnson, D. W., & Johnson, R. (1999). Learning together and alone: Cooperative, competitive, and individualistic learning (5th ed.). Boston: Allyn and Bacon.

Kollar, I., Fischer, F., & Hesse, F. W. (2003). Cooperation scripts for computer-supported collaborative learning. In B. Wasson, R. Baggetun, U. Hoppe, & S. Ludvigsen (Eds.), Proceedings of the International Conference on Computer Support for Collaborative Learning - CSCL 2003, COMMUNITY EVENTS - Communication and Interaction (pp. 59-61). Bergen, NO: InterMedia.

Linn, M. C. (in press). The Knowledge Integration Perspective on Learning and Instruction. In Sawyer, R. Keith (Ed.), Cambridge Handbook for the Learning Sciences. Cambridge, MA: Cambridge University Press.

Reisig, W. (1980). Petri nets: An introduction. Springer-Verlag:NY

Sipser, M. (1997). Introduction to the theory of computation. International Thompson Publishing:Boston.

Slavin, R.E. (1980). Cooperative learning. *Review of Educational Research*, 50(2):315-342.

Slavin, R.E. (1996). Research on Cooperative Learning and Achievement: What we know, what we need to know. *Contemporary Educational Psychology*, 21(1):43-69.

Wolfram, S. (2002). *A new kind of science.* Wolfram Media: Champagne, IL.

Wyckoff, P., McLaughry, S., Lehman, T., & Ford, D. (1998). T Spaces. *IBM Systems Journal, 37*(3), 454-474.

Zurita, G., Nussbaum, M., & Sharples, M. (2003). Encouraging face-to-face collaborative learning through the use of handheld computers in the classroom. In: Mobile HCI 2003, Fifth International Symposium on Human Computer Interaction with Mobile Devices and Services. Udine, Italy.

Figures

Figure 1: Users can create annotations in their private board and then drag them to the public board where others can reposition them or drag them back to their own private board.

Figure 2: Self-assessment of students before (activity 1, left) and after the session (improvised activity on the same board as in activity 1, right)

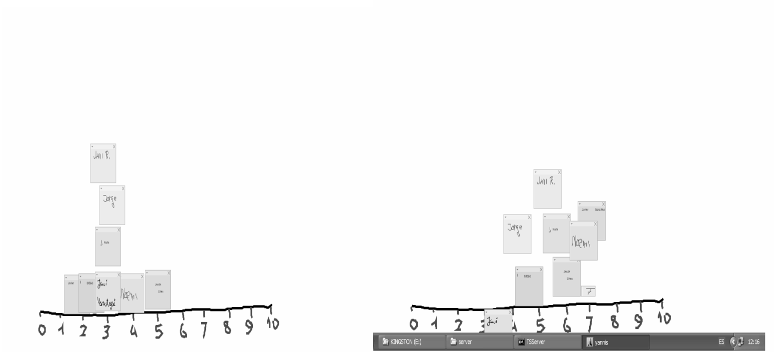


Figure 3. A planned activity flow diagram included at the public board of GS. Note the awareness stickers during the experts' activity (jigsaw CLFP) of the analysis phase.

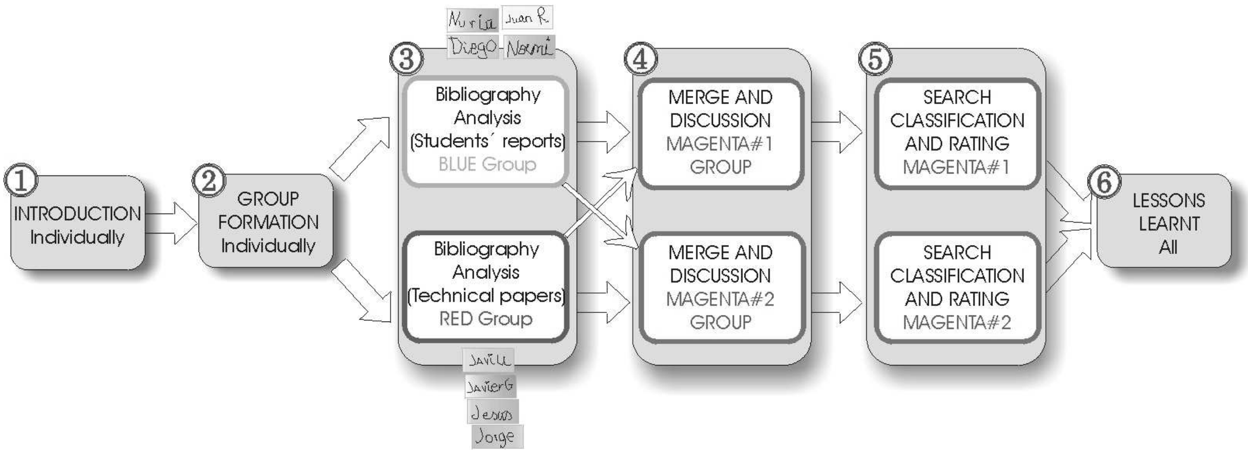


Figure 4: Client views of GroupScribble implementation of the sequencer game in play.

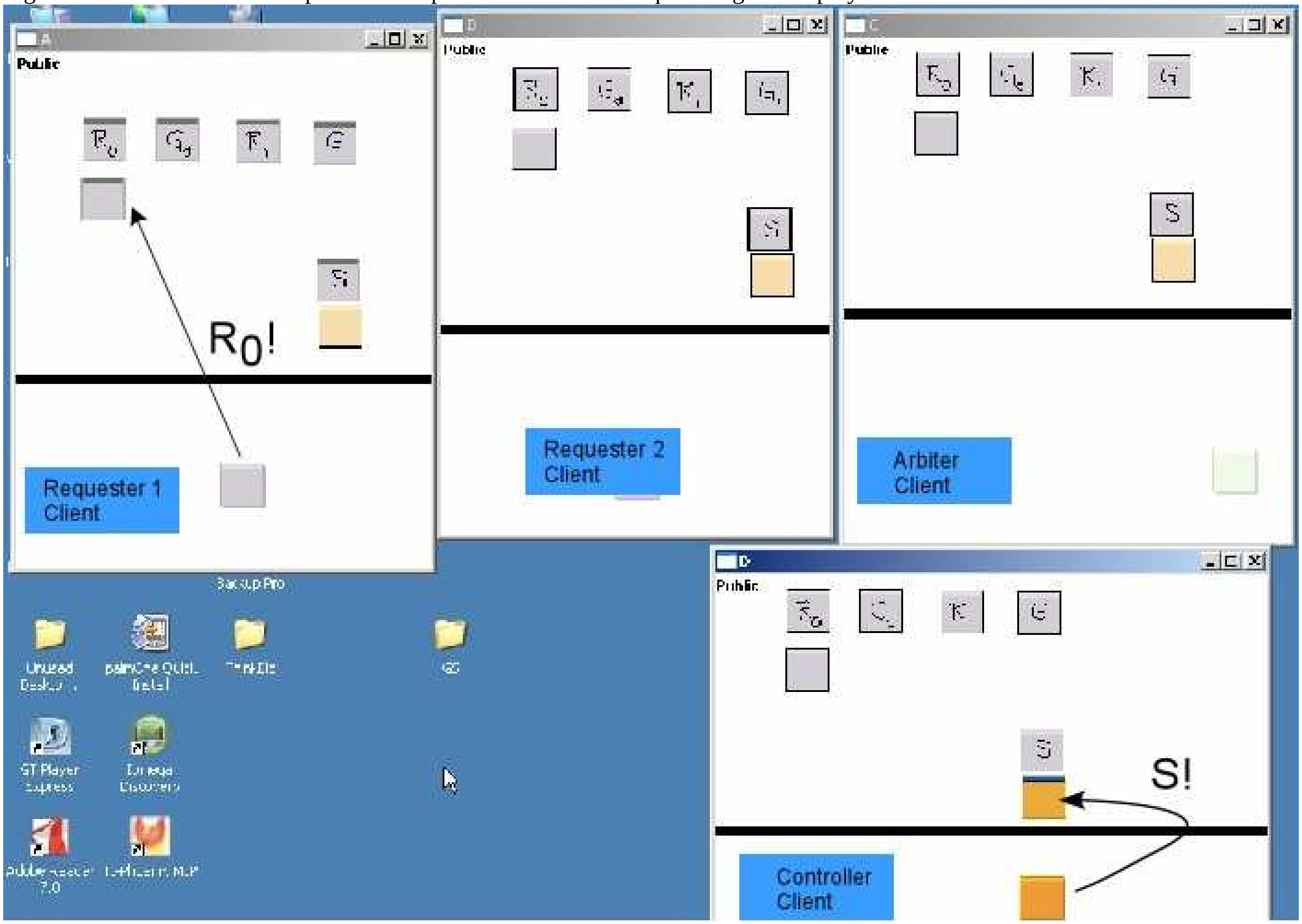
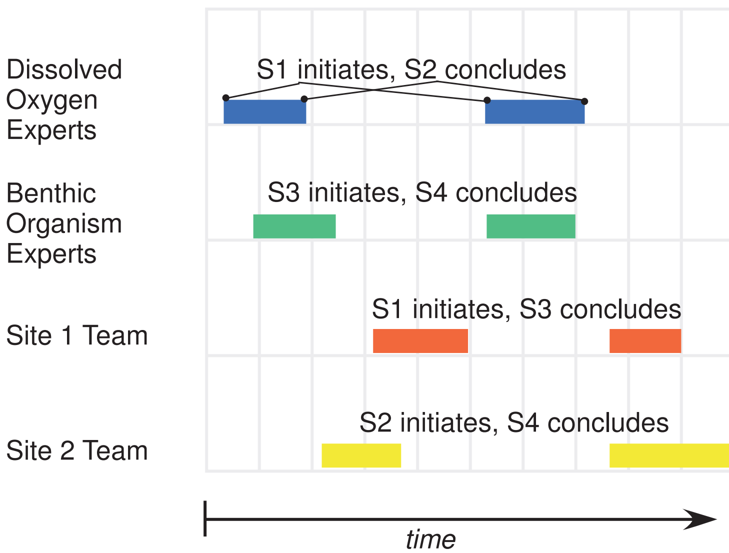


Figure 5: Chronological ordering of two cycles of meetings in accordance with the lead students' pattern.



ERROR: undefined
OFFENDING COMMAND:

STACK: