International Journal of Computer-Supported Collaborative Learning (2019) 14:325–351
https://doi.org/10.1007/s11412-019-09306-1

Supporting classroom orchestration with real-time feedback: A role for teacher dashboards and real-time agents

Mike Tissenbaum[1] · **Jim Slotta**[2]

Received: 2 August 2018 / Accepted: 5 September 2019 / Published online: 24 October 2019

Abstract
This paper investigates the role of the physical classroom environment, coupled with a technology environment that includes real-time agents and data analytics, to support the orchestration of complex collaborative inquiry designs in a high school physics classroom. This design-based research contributes to the wider domain of scripting and orchestration (e.g., Dillenbourg 2012; Dimitriadis 2012; Fischer et al. 2013). Guided by a theoretical perspective of learning in knowledge communities (Slotta et al., 2018), we partnered with a physics teacher to co-design curricular activities and assessments that engaged students in collectively solving, tagging and evaluating physics problems, creating a knowledge base of student-contributed examples, and using those examples as a resource in collaborative inquiry challenges. To support the teacher in orchestrating such a complex curricular design, we developed a tablet application that allowed the teacher see the state of the class in real-time, control the flow of activities and helped him know when and where he was needed within the flow of class activities. The tablet leveraged a set of specially designed real-time software agents to process student interactions in real time, allowing dynamic orchestration of student groups, material allocation, and teacher notifications. The paper begins with a review of recent literature on scripting and orchestration, drawing connection to the theoretical perspective of knowledge communities. We then describe our theoretical model, the design-based method, and details of our curriculum and technology environment. The paper concludes with a summary of how the teacher tablet and the real-time software agents helped support the teacher's real-time facilitation and orchestration.

Keywords Learning · Collaboration · Orchestration · Teacher dashboards

✉ Mike Tissenbaum
miketissenbaum@gmail.com

Jim Slotta
jslotta@gmail.com

Extended author information available on the last page of the article

Introduction

Many scholars have advocated for learning designs that build on socioscientific issues, support twenty-first century learning, and connect learners across formal and informal learning contexts. New media and technologies open the door to the design of powerful social forms of interaction, including Web 2.0 aggregation of user-contributed content, social tagging, voting, and collaborative editing (e.g., wikis). Theoretical work in CSCL has also proceeded, defining "collaboration scripts" (Fischer et al. 2013), and the orchestration of such scripts in technology rich environments (Dillenbourg and Jermann 2007; Schwarz et al. 2018; Tchounikine 2016).

The research described here builds on the notion of learning communities, in which all students in a classroom work collectively to develop a knowledge base that can serve as a resource for their further inquiry. We apply The Knowledge Community and Inquiry (KCI) model to specify a complex collaboration script where students are assigned to a progressive sequence of groups (e.g., jigsaw), with context-sensitive materials, real-time collaboration amongst students (e.g., co-editing documents, jointly voting, and tagging), and dynamic, "emergent" representations of student ideas and resources. These elements can be presented and orchestrated across a wide range of devices (laptops, tablets), displays (surfaces, walls, tables) and other interactive media. The presence, for example, of large projected displays can serve as a vital reference for teacher-led discourse about class progress (Tissenbaum and Slotta 2019).

We applied the KCI model to develop a high school physics curriculum, leveraging a range of technologies and learning analytic approaches to orchestrate students in their assignments to groups, allocation of materials and activities, and collection and aggregation of resources. Forming a co-design partnership with the teacher (Roschelle et al. 2006) we designed a semester-length course in which students developed a sophisticated web of user-contributed content that was socially and semantically tagged, serving as a source of materials for subsequent inquiry activities and informing the large, dynamic displays of their emergent knowledge.

We begin with a review of the literature surrounding collective inquiry and learning communities, including the role of scripting and orchestration, and identify a possible role for real-time software agents as a means of orchestrational support. We follow this with the description of our curriculum, focusing on the co-designed culminating smart classroom activity, and the technology framework we developed to support its enactment, called SAIL Smart Space (S3). We then analyze S3 and its software architecture in terms of its ability to support the enactment and orchestration of real-time inquiry activities, with a focus on the tablet-based real-time teacher dashboard. We conclude with a discussion of the role played by real-time agents and other data-driven orchestration supports within our knowledge community and inquiry curriculum.

Collective inquiry and learning communities

One promising approach to the design of active learning is to consider the entire classroom as a learning community (i.e., as opposed to each student learning independently). In its most simple form, this occurs whenever an instructor asks for a show of hands, or uses a clicker-system to show students how their opinions on some problems may be distributed within the community. A more elaborate application of this approach involves user-contributed content, where the whole class is asked to contribute resources to form a collective knowledge base, such as a Pinterest board, a wiki, or a Google Doc. In this approach, *each student feels as if they are contributing something to a larger corpus that will be consequential for the* *community's progress*. Engaging students in contributing, curating and applying their own content to inquiry projects is a daunting challenge for educators – even those who are experienced in inquiry-oriented methods.

In a learning community approach, students bring their diverse interests and expertise to some common goal. They must all hold a shared understanding that their learning activities will align to advance the community's cause while at the same time helping individuals learn, and allowing everyone to benefit from the community's resources (Bielaczyc et al. 2006). In a review of learning community models, Slotta and Najafi (2013) articulated three common characteristics: (1) an epistemic commitment to collective advancement, (2) a shared community knowledge base, and (3) common modes of discourse. Several scholars have observed that it is challenging for teachers or researchers to coordinate a learning community approach (van Aalst and Chan 2007). As observed by Kling and Courtright (2003, p. 221), "developing a group into a community is a major accomplishment that requires special processes and practices, and the experience is often both frustrating and satisfying for the participants". The limited success or uptake of this approach has been due to the pragmatic and epistemic challenges of shifting from a didactic mode of "knowledge transmission" into one of collective inquiry. But it is also due to the lack of explicit models to guide the design of curriculum where students are interconnected in a progression of individual, small group and whole class activities, creating and consuming materials from a community knowledge base (Slotta and Peters 2008).

The Knowledge Community and Inquiry (KCI) model (Fig. 1) guides the design of science curricula in which the whole class (or even multiple class sections) work together, with all students held accountable for content learning gains (Slotta and Peters 2008). The model includes principled requirements for (1) a knowledge base that is indexed to the targeted science domain (2) collective, collaborative and individual inquiry activities in which students co-construct the knowledge base and then use it as a resource for further inquiry, and (3) assessable learning outcomes that allow teachers to evaluate student progress. The teacher has a scripted role within a KCI design, but also plays a general orchestration role, with aid from a technology environment that coordinates group assignments, material allocation, aggregation of content into "emerging learning objects," and real-time processing of student interactions.

Within KCI curriculum, students are typically engaged in computer-supported inquiry activities, including note taking, observations, brainstorms, problem solving, modeling and simulation, design and argumentation. Prior KCI research has developed sophisticated server software known as the Scalable Architecture for Interactive Learning (SAIL) that captures student contributions (i.e., the knowledge base), and client applications for students and teachers that support the collection, distribution, curation and application of that content. This software infrastructure, collectively known as SAIL Smart Space (S3) provides a flexible foundation for collective inquiry, and was extended and applied in the proposed work, supporting (1) the development of real-time agents that influence student grouping and the distribution of materials; (2) the application of large, dynamic displays (e.g., projectors or smart boards) of the community's emergent knowledge in influencing discourse; and (3) teacher orchestration tools, including representations of the state of the class and flow-control applications.

Scripting in CSCL activities

Curricular designs that engage students as a learning community and integrate rich inquiry and technology environments are likely to be more complex and dynamic then in previous generations of CSCL. Increasingly, designs will need to include the configuration (and

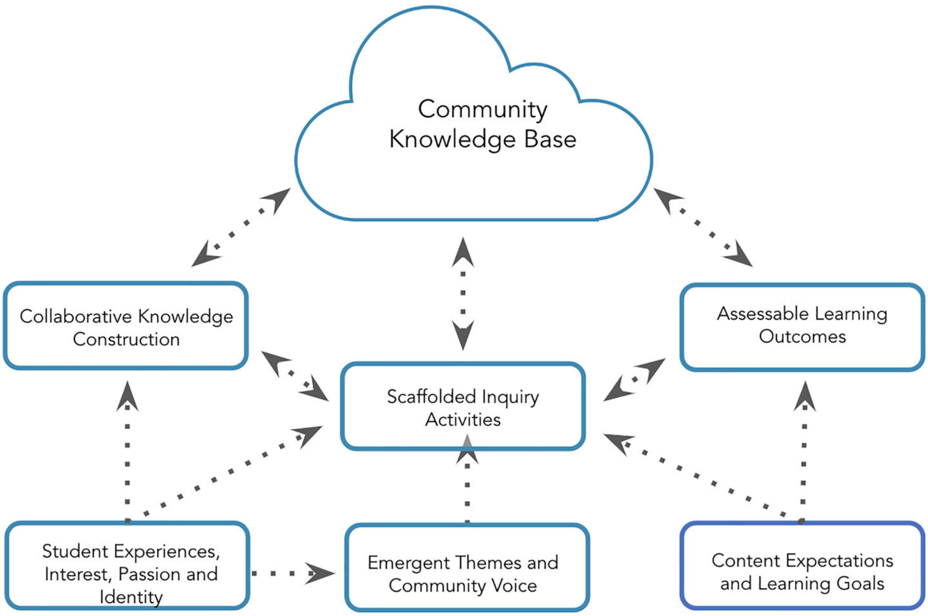

Fig. 1 The KCI Model

potentially dynamic reconfiguration based on emergent patterns) of student groups and activities, teacher roles, and technologies. In order to deal with this complexity, many researchers have advocated for the development of pedagogical "scripts" that can help guide students through complex inquiry tasks by segmenting the learning into more cognitively manageable phases and providing instruction on the formation of groups, distribution of roles, phases of work, and expected deliverables (Kirschner et al. 2004; Dillenbourg 2002).

Inquiry-based scripts often span multiple class sessions, consuming weeks or even months of curriculum time, and thus need to accommodate multiple scales of time, student configuration, and contexts (Lemke 2000; White 2018). In order to respond to the varying granularities of a curriculum (i.e., across space and time, as well as other variables), designers need to think both in terms of the macro script, which describes the overall goals and timing of individual activities (e.g., a field-trip, or a homework task), as well as the finer grain scripts, which specify the individual homework items, student work groups, materials, tools, and scaffolds (Tissenbaum and Slotta 2019).

Of particular interest within and across such scripts, are the granularities of student collaboration. Dillenbourg and Jermann (2007) define five general grain sizes of activity: the individual phase; the group phase; the class phase; the community phase (influencing peers outside one's classroom); and the world phase (contributions to the wider public). In designing curricular scripts, it is critical to ensure the granularity of the task (i.e., the work to be done) matches the granularity of the group size, as a poor match can significantly hinder student learning (Lemke 2000; White 2018).

Within a script, group configuration serves to formalize student roles. Students may alternate in roles such as presenter, discussion leader, moderator, or devil's advocate (Soller 2001; Palincsar and Herrenkohl 2002). Scripting group configurations can allow different materials to be distributed amongst group members, reducing the need for every student to have all the knowledge "in their own skulls" (Hollan et al. 2000), or to require collaboration in order for them to complete tasks (similar to Jigsaw groups – Aronson, 1978). For instance, in *Alien Contact* (Dunleavy et al. 2009), each student in a group was assigned a role (Chemist, Cryptologist, Computer Hacker, and FBI Agent). Depending on their role, each member received different data on their handheld device about an alien artifact (e.g., a spaceship wing), which they had to share in order to determine its significance. This distribution of roles and information fostered positive interdependence and cross-disciplinary knowledge sharing and higher-order thinking skills amongst group members.

Some researchers argue that there is a need to understand the varying strengths, weaknesses, background knowledge, and interests of students when configuring groups in order to ensure productive outcomes (O'Donnell and Dansereau 1992). At the outset of many inquiry curricula, such detailed information on individual group members may not be easily available, only coming to light during the curriculum's enactment. It may therefore be necessary to capture and process information on individual and group performance (either by the teacher or the system itself) to enable adaptive group or material assignments. Technology can play a vital role in this regard, as requiring teachers to process large amounts of interactional data (e.g., responses to assessments, preferences, or patterns of engagement) on their own would be prohibitive (Tissenbaum et al. 2012).

Orchestration of scripted activities

As described above, complex inquiry scripts – especially those involving technology environments and real-time or adaptive conditions – can place a heavy load on teachers, requiring them to simultaneously organize materials, assign student roles and groups, and track individual, group, and whole class progression through activities (Dimitriadis 2012). Several scholars have advanced the notion of Orchestration to define the enactment of such scripts in both the short- and long-term, across multiple contexts and social levels (e.g., Dillenbourg et al. 2009). Whereas scripting deals with the structuring of activities *before they are enacted*, orchestration is concerned with the regulation of an activity *once it has begun* (Soller 2001). Orchestration introduces a level of flexibility to the execution of a script, allowing for the "re-scripting" of groups, student roles, materials presented, and even which steps come next. This is especially important in inquiry-based curricula, which often require the ability to adapt in response to emergent class patterns, community voices, or new and interesting avenues for investigation.

Orchestration places the teacher at the center of the learning process as a "conductor," orchestrating a broad range of activities (Kollar et al. 2011). Rather than as a knowledge provider, the teacher is responsible for making timely and context-relevant adjustments to the script based on assessments of individual and whole class progress, collaboration and growth of ideas (Sharples 2013). While this support of activity progression and resource distribution could theoretically be done without technology supports, many have argued that technology environments can make the process "smoother" and reduce the teacher's "orchestrational load" – particularly in scripts that require the tracking of every student in the class and their individual resource needs (Dillenbourg 2012; Nussbaum et al. 2009).

Technologies that work in support of the orchestration of classroom activities generally fall under one of two complementary forms: Orchestration Technologies and Orchestrable Technologies (Tchounikine 2013). Orchestration technologies directly support the teacher in managing curricular activities (Dillenbourg et al., 2011). In *Edunova* (Roschelle et al. 2010), for example, students are sent fractions problems on their handheld devices to collaboratively solve in small groups. As groups submit answers, the teacher (on his or her personal device) sees a color-coded matrix letting him know which students got the answer right on the first try (green), within a specified number of tries (yellow), or if they failed to get the answer right within a specified number of tries (red). Given this detailed information the teacher can enact formative assessments and adapt his actions in response to specific student needs. A similar approach can be seen with Texas Instruments Nspire Navigator system, which allows the teacher to control the flow of the class through actions such as beaming a student's screen to the front of the class or sending quizzes to students' calculators (Clark-Wilson 2010). In this case the teacher can use the system to generate formative assessment of the class' knowledge and adapt the orchestration of the classroom activities.

Orchestrable technologies are those whose precise function can be determined or adapted both before and during an activity. In some cases, orchestrable technologies can add a layer of flexibility to the script by allowing for fine-tuning or real-time adapting of the script by teachers, students, or the system itself. For instance, in EvoRoom (Lui and Slotta 2014), students are immersed in a simulated rainforest as they conduct investigations about flora and fauna. The teacher is equipped with an "orchestration tablet" that allows her to advance or retreat the date of the simulation across millions of years, depending on the kinds of habitat and ecology she wants the students to investigate. In this way, the teacher can adapt the conditions of the classroom in response to emergent class patterns, questions, or inquiry needs.

What is critically important in the examples above is that they provide specific insight into the state of the class, without requiring that the teacher (or TA) take any specific action. Rather, the technologies simply provided information to help them make decisions. Other orchestration technologies may have a more direct role in controlling the flow of activities. Cognitive Tutors, for example, (e.g., Anderson et al., 1995) employ student models to provide timely prompts and progress students through activities based on their past work, freeing up teachers to help those students most in need. However, such fully automated systems have been shown to be prone to "gaming the system" and other off-task behaviors (Baker et al. 2004).

Software agents

With the ability to capture and process data from students' interactions within technology environments in real time, important patterns or insights can be made invisible that would otherwise be too time consuming for teachers to compile on their own. One form of technology that can serve an orchestrating role includes "software agents" – small, active software elements that respond to pre-specified contexts or conditions, process the actions or interactions of participants, performing a kind of real-time data mining (Serenko and Detlor 2002), and operating on semantic metadata (Brusilovsky, 2001). In addition to their use in education (Serenko and Detlor 2002; Yau et al. 2003), software agents have seen significant growth in recent years across multiple sectors including business and e-commerce (Papazoglou 2001; Jennings 2001), health (Abowd and Mynatt 2004; Cook and Das 2007), air traffic control (Wooldridge and Jennings 1995), and video games (Stanley et al. 2005). What separates agents from traditional software is that agents are capable of responding to the state of their environment and conducting *flexible* autonomous actions in order to meet their design objectives (Jennings & Wooldridge, 1998).

O'Driscoll et al. (2008) state that for educational settings, agents need to be particularly aware of the *context* in which the learning takes place, the identities of nearby people and objects, the social setting (individual, small group, or whole class settings), the specific activity being performed, and that an agent must ideally be able to adapt according to its context, including to any changes that might happen to these various factors over time. Within these contexts, agents can capture individual and whole class learning traces, generated artifacts, and emergent metadata to provide new insights and supports for student learning (Roschelle et al. 2013).

Software agents thus hold promise for the design of scaffolding environments to support inquiry learning, in part because they allow orchestration of scripts that are deliberately *ill determined* (i.e., scripts where it is not known, in advance of the enactment, what outcomes or conditions will emerge from the products of student interactions). The use of agents allows for open-ended designs, enabling the script to evolve in relation to student interactions. For example, students might be engaged as a learning community to understand environmental conditions in their neighborhoods. Agents could identify two students who independently looked up CO_2 sensors and then suggest they share notes or work together to advance their understanding. Agents could then dynamically re-group these students with peers they hadn't worked with previously to combine their ideas with those of the larger class. The core idea here is that agents can respond to a wide spectrum of conditions *as they emerge* – most of which would be operationally impossible for a teacher to do on his own.

Supporting the teacher as a facilitator

The goal of smart classrooms and agent driven orchestration should be to engage teachers as active co-participants and facilitators of student learning, rather than relegating them to "guides on the side" (Pea and Maldonado 2006). The idea of the teacher as a "wandering facilitator" has been advanced by Hmelo-Silver's (2000) as a paradigm for supporting learning in student-driven inquiry designs. In the wandering facilitator model, the facilitator rotates from group to group, adjusting the time spent with each of the groups in the classroom according to their needs (Hmelo-Silver 2004).

However, supporting a teacher as a wandering facilitator is a persistent challenge (Hmelo-Silver 2004), as it requires the teacher to be aware of each group's state within the flow of activities and where she is most critically needed. Adding additional informational cues (Alavi et al. 2009) and real-time agents, can reduce the orchestrational load placed on teachers and help them make better informed decisions about where they are needed in real-time. For instance, Schwarz et al. (2018) showed how the use of machine learning could provide teachers alerts when small groups were engaged in *critical moments* during a collaborative geometry class. Making these critical moments visible for the teacher provided insight into where and when they were needed, helping them orchestrate the class' conceptual learning.

Given the rapidly growing spectrum of data that *can* be provided to teachers, we need to make sure the information we provide is useful, timely, and actionable. Providing extraneous information or "noise" can actually increase orchestrational load and, in fact, become a hindrance to effective teacher facilitation.

Research questions

In response to these ideas, with a focus on the design and implementation of a tablet to support teacher classroom orchestration, we investigate two central research questions: (1) How can the teacher tablet leverage real-time agents to support the orchestration of collective inquiry, including context sensitive material assignment, appropriate student grouping, and coordination of inquiry activities? And, (2) How can the real-time information provided on the tablet help a teacher orchestrate class activities?

In answering these questions, it is important to note that this work focuses on how the teacher tablet supported the orchestration of the overall curriculum, rather than student learning and interactions. With this said, the ability of the environment to successfully enact the kinds of complex pedagogy described below and support the teacher in orchestrating classroom activities, has been described as a grand challenge within the CSCL community. As such, this paper plays an important role in advancing research into this area.

In the following section, we outline our design-based research method, in which a twelve-week KCI curricular intervention was developed for two Grade 11 physics classes. We describe a smart classroom framework developed to support our orchestration, including the role real-time agents, and our analytic approach for evaluating our design in term of how those features supported the orchestration of our curriculum.

Material and methods

Co-design and design-based research

A design-based research approach (DBRC 2003) was employed for this study, which built upon several earlier design cycles (see Tissenbaum and Slotta 2019). Rather than validating a particular curriculum, the central goal of design-based research is to advance a set of theories on learning that transcend any particular design or enactment (Barab and Squire 2004). To this end, the primary outcome of this research is the design and evaluation of the technological and orchestrational infrastructures themselves (rather than any particular student outcomes), with the aim of understanding their role in supporting complex collective inquiry activities.

Even when well designed, technology-enhanced learning environments can be quite challenging for a teacher to integrate into her everyday classroom practice (Slotta and Linn 2009). Success can be heavily dependent on how well the teacher perceives the "fit" between the intervention and his or her goals for students, teaching strategies, and expectations for student learning (Roschelle et al. 2006). As the complexity of the learning design increases (e.g., in a KCI learning community approach, which can entail substantive commitment to collective and collaborative inquiry designs), the teacher will be increasingly challenged to integrate all the elements successfully – even if she was an active participant in the curriculum design. We employed a co-design methodology (Penuel, Roschelle & Shechtman, 2007), in which the teacher was engaged as an active participant in the curriculum and technology designs to ensure that our innovations fit within his content expectations and goals for student learning. The current design builds on several earlier iterations within the same classroom, which together have addressed the notion of a "smart classroom" infrastructure for supporting collective inquiry (Tissenbaum and Slotta 2019).

Participants

This study involved two grade-eleven physics classes ($n = 22$, $n = 23$), in a fee-based, high achieving urban high school in a major metropolitan city. The same teacher (a science teacher with over 10 years of teaching experience) taught both classes and was the co-design partner from earlier smart classroom studies spanning the previous two years (for a detailed description of the iterative technology and room development see Tissenbaum and Slotta 2019). While this setting may not reflect all the circumstances of public-school settings, it is an appropriate context for our research, which entails complex designs with many different technologies and a high level of autonomous inquiry from students. We do anticipate extending these approaches to support a wider range of contexts, which is addressed in our discussion.

Technology infrastructure: SAIL smart space (S3)

Our designs required a flexible and adaptive technology infrastructure that could support the orchestration of collaborative activities including spatial, social, and semantic dependencies. In response, we developed SAIL Smart Space (S3 – Fig. 2), an open source framework that can capture the products of student inquiry (e.g., notes, votes, or tags), the coordination of complex pedagogical sequences, including dynamic sorting and grouping of students, and the delivery of materials from the knowledge base based on emergent semantic connections.

An important goal in developing S3 was to allow the physical space of the learning environment to play a meaningful role within the learning design – either through locational mapping of pedagogical elements (e.g., different locations in the room are scripted to focus student interactions of different elements or topics of inquiry) or through orchestrational

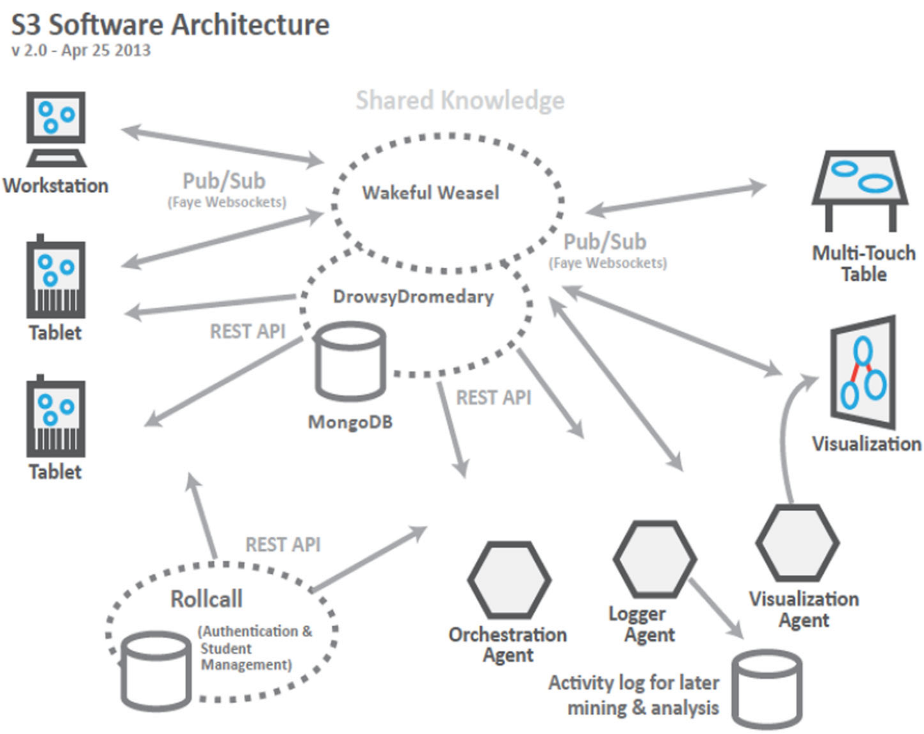

Fig. 2 SAIL Smart Space (S3) systems architecture, showing the use of direct WebSocket messaging to enable communications amongst any element of the environment, a persistent, non-relational (no SQL) database (MongoDB) and software agents

support (e.g., physical elements of the space, like projected displays, help guide or coordinate student movement, collaboration, or activities).

We also added a layer of intelligence to our learning environment through the addition of real-time data mining and computation performed by software agents. Because of the complex nature of our KCI designs and the high demand they place on teachers (as described above), we felt that such agents could play an important role in support of student inquiry and in reduce the teacher's orchestrational load, by automating some tasks and helping the teacher make more informed orchestrational decisions.

Our design improvements to S3 also included ambient displays that were coupled with the community's emerging knowledge and in-the-moment activities such that they provide a source of peripheral information for students and teachers alike (i.e., about time remaining on tasks, or progress in the knowledge base).

S3 comprised a suite of five core technologies: (1) a portal for student accounts and software application management; (2) and software agent framework for data mining and tracking of interaction in real-time; (3) a central database that houses the designed curriculum and products of student interactions; (4) a visualization layer that controls how materials are presented to students across a range of devices and displays (e.g., tablets, laptops, interactive tabletops and large format displays); and (5) a communication framework for connecting materials in the knowledge base (e.g., student notes, class polls, or multi-media) and tangible and physical inputs (e.g, through Arduino micro-controllers) in real-time.

Real-time software agents

As described above, an important new component of S3 was the development of real-time agents to support the orchestration of inquiry activities that included real-time allocation of materials, assignment to groups, or feedback to the teacher. We included four distinct types of agents, as outlined in Table 1.

An important feature of the S3 agents is that they work in concert with each other to create ecologies of orchestration (i.e., nested conditions that feed into each other to allow for interdependent decisions and orchestrational moves). As part of our description of the curricular intervention, we outline several of these ecologies. We follow this with an evaluation of their support for classroom orchestration.

Developing an inquiry script – PLACE

In order for the smart classroom activity to be more than just supplemental in nature, we needed to develop a complete curriculum in which the smart classroom was one of several learning contexts, integrated within a broader progression of activities across classroom and home settings. In order to investigate how the smart classroom could leverage student-contributed content for purposes of authentic inquiry activities, we required a script in which students produced artifacts that would be meaningfully reused in successive activities. We therefore designed a KCI-based physics curriculum in which smart classroom technologies supported collaborative and collective forms of inquiry for students, and supported critical reflection and formative interventions for the teacher.

The teacher shared two main goals for his course: first, to help students recognize "physics in their everyday lives" and bring this view of physics back into traditional classroom settings; second, for students to develop a *coherent understanding* of the

Table 1 S3 Real-time Software Agents used in S3

Student Sorting Agent	•Sorts students both into groups and around the room •Sorts can be designed in two ways: ○Pre-set by the instructor or researcher ○Emergent based on individual, small group, or whole class actions
Consensus Agent	•Monitors groups of students where activities require achieving consensus ○Students cannot move to the next step until consensus is achieved •Also used as an orchestration tool to alert teacher to review student consensus when necessary
Bucket Agent	•Coordinates the distribution of materials to students in two possible ways: ○Ensure that all members within a group had an equal but unique subset of materials from a given set (i.e. a series or problems or equations) or ○Distributed materials to all members to ensure reduce the variance between members completing a task (quicker students may receive more items to work on than slower students)
Student Progress Agent	•Tracks individual, small group, and whole class progress •Sends updates to other devices (i.e. ambient display, teacher tablet) ○Can aid both teacher and students in knowing if students are falling behind the rest of the class •Coordinates the timing and delivery of materials

underlying principles of the course, including the connections amongst those physics principles (i.e., to "see that all the principles are tied together"). These goals aligned with the regional curricular guidelines for grade-11 physics: (1) use the appropriate scientific models to explain and predict the behavior of natural phenomena; (2) analyze and synthesize information for the purpose of identifying problems for inquiry, and solve the problems using a variety of problem solving skills; and (3) locate, select, analyze, and integrate information on topics under study, working both independently and as part of a team (Ontario Ministry of Education 2008).

In response to the second goal, we began by generating set of fourteen "core" principles (Table 2) that the teacher felt were of core relevance to the course.

We then co-designed a 12-week curriculum called PLACE (Physics Learning Across Contexts and Environments), which engaged students in capturing examples of physics in the world around them (through pictures, videos, or open narratives), and then using those examples as a source of inquiry – generating problems, applying conceptual tags, and using them as examples. The products of these various inquiry activities became a dynamic "community knowledge base" (one of the central features of KCI) that evolved from one unit to the next. This knowledge base, called PLACE.web, served as a resource for the culminating smart classroom activity, in which students applied what they had learned across all three units to solve ill-structured physics problems relating to scenes from popular Hollywood movies. The smart classroom served as the technology enhanced environment in which we address the research questions articulated above (i.e., within the Smart classroom environment).

Table 2 Grade 11 Fundamental Principles

Newton's First Law	Acceleration	Fnet = 0	Kinetic Energy
Newton's Second Law	Uniform Motion	Fnet = Constant (non-zero)	Potential Energy
Newton's Third Law	Kinetic Friction	Fnet = non-constant	Conservation of Energy
	Static Friction	Vectors	

Culminating smart classroom activity

As a culminating activity in the PLACE design, we created a challenging task in which students analyzed the physics contained within several popular Hollywood movie clips, in order to test their validity of the scenes. This culminating activity centered around the Smart Classroom, involved three short scripts that spanned home, a traditional class setting, and a smart classroom, and relied heavily on S3 agents to coordinate the distribution of materials, roles, and tasks.

At home activity At home, students were tasked with looking at a collection of the problems they had been assigned during the preceding 12-weeks (including their own contributed challenge problems and new problems developed by the teacher), verifying their tagging of relevant physics principles, and adding equations that might be used to solve the problems.

Classroom activity In-class, students worked in small groups, using tablet computers to reach consensus on a refined “final set” of the tags and equations for each problem. The goal of this activity was for students to achieve consensus about the principles and equations that had been assigned to each problem in the corpus. The group was assigned one of the problems, with each student seeing the problem and its various tags on his or her tablet (from the individual at-home activity), and asked to agree or disagree. The group was required to reach consensus on all of the principles and equations before they could move to the next problem. Achieving consensus is an important task for students, as it provided opportunities for student to clarify concepts and understanding, towards gradually improving their knowledge through sharing and discussion (Purba and Hwang 2017). Students could see the work of their group members in real-time, reflected on their own tablets, which helped facilitate face-to-face discussions. The resulting set of problems, tagged with principles and equations, was then stored in the knowledge base for use within the final smart classroom activity.

Main activity: In the smart classroom For the third and final stage of the culminating script, we developed a set of tools that took advantage of the physical and collaborative affordances of the smart classroom, including large projected displays accompanying each station, and individual tablet computers to support students as they performed activities. Both classes were split into two smaller sections of 11 or 12 students, with each section engaging in the smart classroom activity on a different day (i.e., 4 days in total). Upon entering the smart classroom, students were engaged in solving a series of ill-structured physics problems using Hollywood movie clips as the domain for their investigations (e.g., could Iron Man Survive a fall to earth, as depicted in the movie?). Four videos were presented to the students, each at a distinct physical location within the room. The students were engaged collectively, working as a whole group of 10–12, as well as collaboratively, in various small group configurations as directed by the S3 real-time agents.

The smart room script was broken up into four different steps as shown in Fig. 3: (1) Principle Tagging; (2) Principle Negotiation and Problem Assignment; (3) Equation Assignment, and Assumption and Variable Development; and (4) Solving and Recording. In each step, students moved from one video to another, completing a set of collective and collaborative tasks that built upon the emerging knowledge base, using tablets and large format interactive displays.

The large-format interactive displays aggregated the products of individual student work from their individual tablet inputs and helped facilitate group discussion (Fig. 4). S3 software agents provided students with context specific tasks and materials, facilitated the dynamic grouping of students, and ensured student consensus on final products was reached on all collaborative tasks. We also developed a set of ambient displays that showed real-time

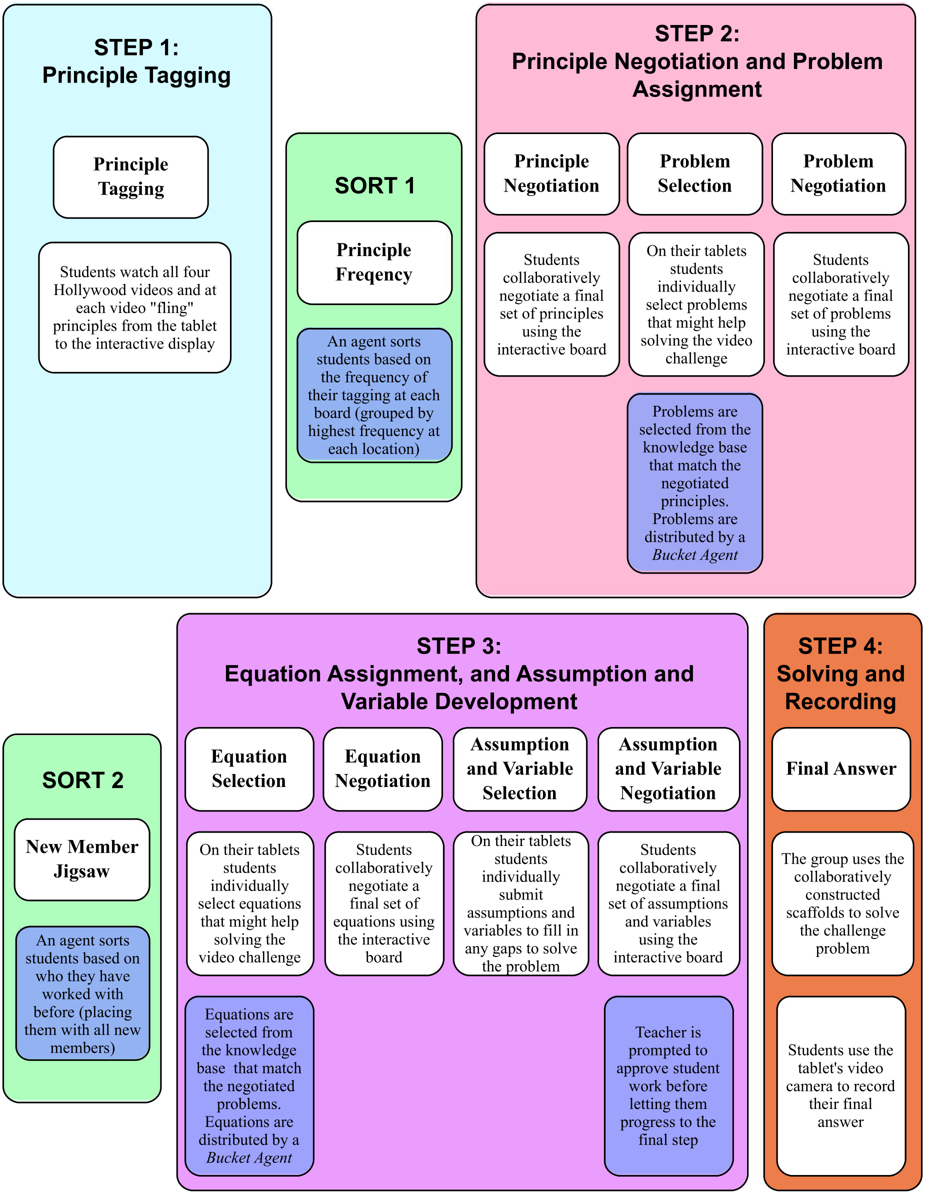

Fig. 3 The smart classroom "Hollywood Physics" script involved four distinct steps. The dark blue boxes indicate actions mediated by real-time software agents. The red box indicates the point in the script where the real-time software agents alerted the teacher to review individual groups' work for approval or to have them go back and refine their thinking

information on the state of class activities and an orchestration tablet that provided the teacher with additional procedural information and control over the progression of class activities. Below, we outline our rationale for each of these technologies with a specific focus on their roles in supporting real-time teacher facilitation and orchestration.

Designs of teacher orchestration supports

In order to support the teacher as a wandering facilitator and to know where and when he was most needed, we developed a specialized teacher orchestration tablet (Fig. 5). The orchestration tablet, iterating on observations and feedback from previous designs (Tissenbaum and Slotta 2019), moved from a device showing student work post hoc (which the teacher was unable to act upon in real-time), to an orchestrational tool that allowed the teacher to more directly orchestrate the flow of activities. The goal of the orchestration tablet was to give the teacher control of class progression at both whole class and small group levels, and to inform him when he was needed at key moments in the script. The tablet showed him which tasks each group had completed (in contrast to the information at the grain size of the individual, available on the large ambient display), alerted him when he needed to review a group's work, and allowed him to easily progress the whole class to the next step in the activity (pressing a button on the tablet would send a signal to the S3 system, which then managed the student groupings, location assignments, and material distribution).

Design of software agents for curriculum orchestration

In order to address our first research question – how the orchestration tablet could leverage real-time agents to support the teacher's real-time orchestration – we designed specific tasks

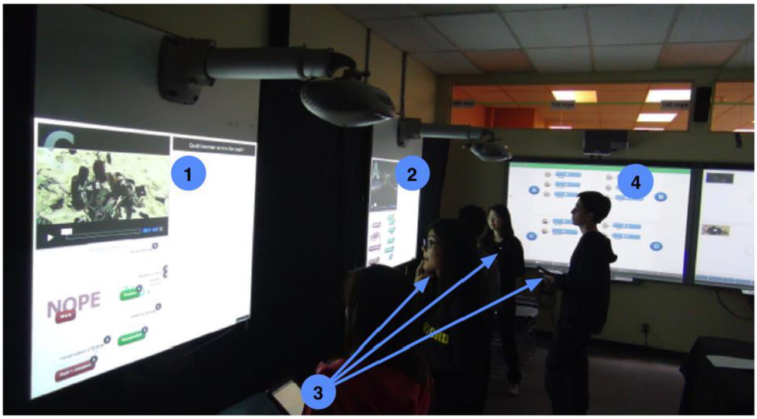

Fig. 4 The smart classroom setting with (1) An interactive collaborative display that orients students towards a specific Hollywood scenario, aggregates student contributions specific to that video and facilitates idea negotiation; (2) A second board with a different scenario facilitates similar but thematically distinct student interactions (two other boards are similarly placed on the opposite wall; (3) Individual tablets provide students task instructions, allow them to access the knowledge base, and contribute ideas to the shared display; and (4) An ambient display that shows where students are in the room, their completed tasks and the time left in the activity

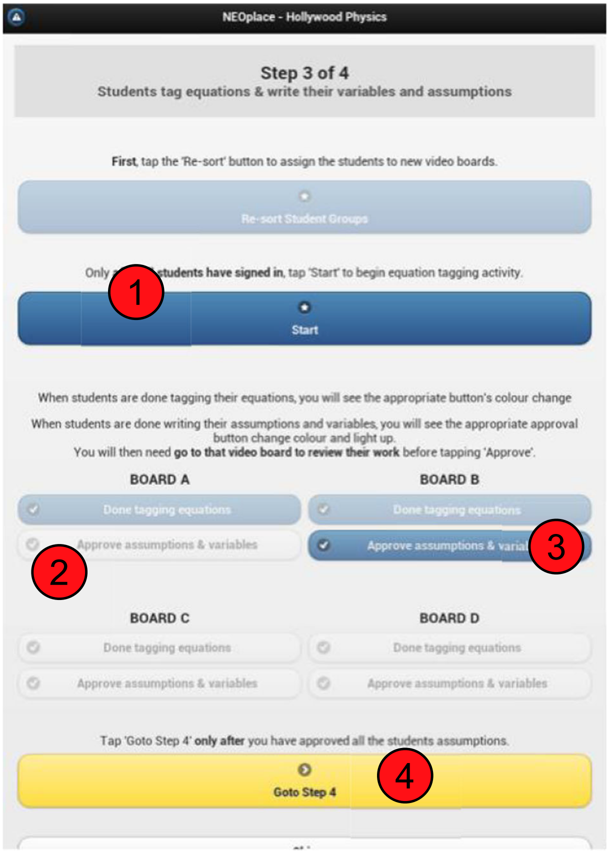

Fig. 5 The Teacher Orchestration Tablet. The tablet (1) Enabled the teacher to start a stage for the whole class; (2) Showed each group's progression through the activity; (3) Alerted the teacher when a group reached a point for intervention (pre-defined by the teacher); and (4) Let the teacher advance the class to the next Step

for the real-time software agents to enact during the activity. Below, we outline these tasks along with the specific agents we developed (described in Table 1 above).

Sorting students based on emergent classroom conditions

Grouping and re-grouping students is a persistent challenge in live classroom settings. This challenge becomes compounded when the conditions for the sorting must emerge during the enactment of the activity itself (and therefore cannot be known a priori). During the smart classroom activity, we wanted students to be sorted based on a set of predefined conditions set in co-design with the teacher: 1) After Step 1, students were sorted based on the frequency of their tags at each of the scenarios in the room, as we felt this might show a particular affinity towards that topic by the student; and 2) After Step 2, as we wanted students to work with students they hadn't worked with in the previous step. Since we didn't know which scenarios each student would tag with which principles beforehand these sorts could only happen in real-time (once the teacher pressed "sort" on his tablet). Once the teacher pressed "sort" on his tablet, the large ambient display at the front of the class showed the students where each of them was to go in the room. In this case only one agent was used:

Sorting Agent: The sorting agent created tables of students' activities (i.e., tags assigned to each scenario, and who each student had worked with previously), and based on these tables, placed students around the room.

Supporting the teacher in just-in-time orchestration

Helping a teacher function as a wandering facilitator within a complex real-time activity takes more than simply making student work visible to them. In order for the teacher to truly react to the just-in-time needs of the students in the class we needed to enable him to know both *when* and *where* he was needed. To understand how the S3 architecture supported the teacher's ability to respond, we had two agents work in concert to alert the teacher when a group reached the end of Step 3 (see Fig. 3 above):

Consensus agent Ensured that the group had sorted all of the variables and assumptions submitted to the negotiation area on the collaborative display. Once completed the consensus agent sent a message to the *Progress Agent*.

Progress agent Once the progress agent received a notification that a group had reached consensus on their variables and assumptions, it then sent a message to the teacher's orchestration tablet, alerting his to review student work – either approving it (allowing them to progress) or have them go back and work on it some more. By using the underlying agent infrastructure and messaging protocol in S3, the orchestrational load placed on the teacher to know (at least on some level) when and where he was needed was reduced by his awareness that he would be alerted on his tablet. This allowed the teacher to more freely roam the room engaging with students based on group needs.

Measures and analytic approach

While it is important to situate the culminating activity within the context of the larger curriculum (in order to show its significance as more than a stand-alone activity), the analysis below will not evaluate the parts of the curriculum that preceded it (for analysis on the preceding activities see Tissenbaum and Slotta 2015). Rather, we restrict our analysis to the enactment and orchestration of the culminating Smart Classroom activity.

As stated above, the main goal of this research was to evaluate the efficacy of S3 teacher tablet and agents to support the teacher's classroom orchestration (rather than evaluating the particulars of student learning). To this end, our measures and analytic approaches focus primarily on evaluating the design in terms of its ability to support the enactment of the designed curriculum, and the role agents played in this enactment.

In order to evaluate and understand the enacted design, we used a mixed-method approach that included multiple data sources to triangulate data and gain a more complete picture of the study (Greene 2006; Mason 2006, Johnson et al., 2007). The use of multiple data sources is particularly relevant in design-based approaches, due to the complexity and innovative nature of their design and enactment (DBRC 2003). Data sources included pre- and post-interviews with teachers and students, sever logs, user contributed artifacts, and audio and video recordings. All user data, generated artifacts, and interaction with the system was collected using S3's data collection infrastructure. Below, we use this data to illustrate key examples of the role the teacher tablet and the real-time agents played in supporting classroom orchestration.

During the culminating activity, six video cameras were used (one at each of four "zones" in the room; one camera with a fixed view of the whole room; and one wandering video camera) to capture student and teacher interactions. To capture student and teacher discourse, voice recorders were place at each of the four zones and the teacher had his own lanyard microphone. Video and sound recordings were synchronized and analyzed using Inqscribe, a popular coding software platform.

Results and discussion

Sorting students based on emergent classroom conditions

We wanted to understand the efficacy of the agents for sorting student based on emergent class patterns. To this end, we examined the server logs to see how the agents sorted the students once the teacher pressed "sort" on the orchestration tablet. In all four class sections, after the teacher pressed "sort", the Sorting Agent successfully sorted students based on their earlier actions in the room. Table 3 (initially reported in Tissenbaum and Slotta 2015) provides data from one section's sorting. The agents used a cascading approach to assign one student to Board A based on their frequency of principles, then one to Boards B, C, and D in order, before repeating this process until all student were sorted. For instance, Jason was assigned to board B and not A, C, or D because the agent had already placed Alice at Board A, and Jason had the most tags when the agent went looking for a Board B student (i.e., for the second assignment by the agent's algorithm).

In our design, the agents used a simple table-based system to decide how to sort students, and we recognize that other approaches could allow for more complex approaches in making such real-time grouping decisions. However, this method was sufficient to demonstrate that the underlying agents were able to track these conditions in real time and make the appropriate decisions as laid out by the teacher. Video analysis of the student sorting noted that the average time from when the teacher pressed "sort" on his tablet to when students were in their new groups, ready to start the next task, was under 20 s. This is noteworthy, compared to what might be achieved in a low-tech classroom setting, where re-grouping based on evidence from a previous activity would take time for the teacher to compute, followed by more time to convey the grouping to students and get them to move around the room. In our case, the automated tracking and assigning of students within groups allowed the teacher to focus on helping the students and not the logistical aspects of the group sorting. This point was reinforced by the teacher's comment in the exit interview:

> Each [agent sort] was a different ensemble, using physics pedagogy and other schemes to figure out where kids should go. During transitions when you're a teacher getting kids up, moving them to different seats – you waste so much class time doing that. Even a common group, cooperative learning scenario, like a games theory thing, where kids are really learning from each other, just getting the kids to move around the classroom adequately for that, I find cumbersome – I just kind of dread moving the kids around the

Table 3 Student Tagging frequencies and Sorting Agent assigned boards

# of Tags by student								
Student	Board A	Board B	Board C	Board D	First sort: sent to board	Second sort: Sent to board	Sorted to new board?	Sorted with new members?
Alice	4	3	3	4	A	B	Y	Y
Pearl	3	0	3	2	A	C	Y	Y
Jason	4	3	4	4	B	C	Y	Y
Rob	0	3	3	1	B	D	Y	Y
Desi	3	2	3	0	C	D	Y	Y
Raffi	0	2	2	2	C	A	Y	Y
Becky	2	2	3	3	D	A	Y	Y
Sun	2	2	0	2	D	B	Y	Y

> class and organizing that, rather than doing the activities themselves, and so I just loved the logistical assistance that [the S3 agents] offered.

Supporting the teacher in just-in-time orchestration

Across all four class sections of the culminating activity, the S3 agent and messaging framework successfully notified the teacher whenever a group required the teacher's review and approval. For each individual group, the *Consensus Agent* was able to pick up when they completed their assumption and variable negotiation (at the end of Step 3), and sent a notification over S3's messaging service. The *Progress Agent* was able to interpret the event as one that required teacher response, and sent the appropriate message to his tablet (Fig. 6).

In total, the teacher was sent 23 alerts to review students work (Table 4). It is worth noting that across the four sections there were only sixteen groups (four in each section). The reason for the seven extra alerts was that the teacher asked six groups to refine their thinking more and resubmit it for review (with one group being asked to re-submit twice). This is important in several ways. The first is that it shows the flexibility of the agents to respond to multiple similar events with the same group, which allowed for a more flexible (rather than a strict linear) progression through the activity. In addition to the orchestrational flexibility these alerts provided the teacher, this approach shows how an orchestration tablet (powered by the software agents) may allow teachers to offload

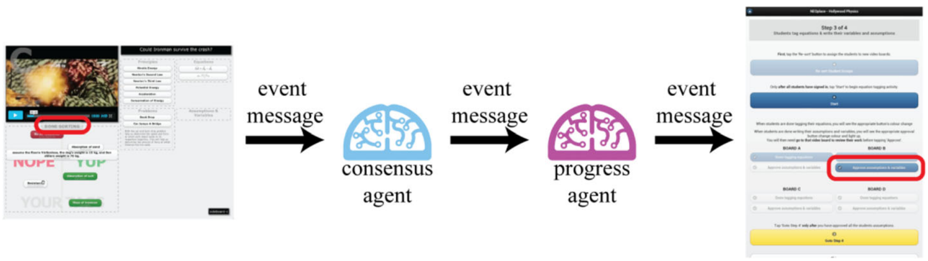

Fig. 6 An example of the event messages handled and sent by the real-time software agents from a group's collective display to the teacher tablet. A *Consensus Agent* would monitor the group's work and wouldn't allow them to submit their work until all the items were sorted. Once the items were sorted and the group pressed submit, a *Progress Agent* would pick up the message and send an alert to the teacher on their tablet

Table 4 Agent-orchestrated alerts for review of student work sent to the teacher's orchestration tablet. Across all four sections the teacher was successfully alerted and reviewed every groups' negotiated set of assumptions and variables

	Zone A	Zone B	Zone C	Zone D
Section 1 Review Alerts	1	2	2	1
Section 2 Review Alerts	1	1	1	1
Section 3 Review Alerts	1	2	1	2
Section 4 Review Alerts	3	1	1	2

the need to constantly monitor the state of student work, instead focusing his attention where they are most needed (as prompted by the tablet).

Understanding the effect of the teacher's just-in-time orchestration

To understand the effect of the teacher alerts and teacher follow-ups on student outcomes, we evaluated the student generated products from that stage of the activity. First, to evaluate the quality of each groups' final constructed set of assumption and variables across all four sections, the teacher (post hoc) scored them using a four-point scale that rated them based on their completeness for setting up the problem (Table 5). Across all four sections, groups averaged 2.6 (out of maximum 3) with no group scoring below a 2, indicating an overall high quality of problem setup.

Next, to understand the effect of the teacher's reviewing and approving of groups' work on their final completeness score, we rescored the original assumptions and variables of the groups the teacher had asked to resubmit (i.e., before their edits). Figure 7 shows changes in groups' completion scores. A paired *t*-test showed that increases in completion scores were significant ($p < .05$) when comparing scores prior to the teacher's intervention (M = 2.17, SD = 0.41) and after (M = 2.83, SD = 0.41; $t = 3.1623$, $p = 0.025$). While the sample size is small ($n = 6$), de Winter (2013) has shown that small-sample *t*-tests are acceptable when assessing changes in student outcomes.

When we examined video of the teacher's interactions with the groups, we found that the teacher largely focused on "teasing out" how the groups came up with their variables and assumptions. For instance, during Session 3, the following exchange shows an interaction in which the teacher asked the group in Zone B to further refine their variables and assumption:

Table 5 Rubric for scoring group assumption and variable construction during Step 3 of the culminating activity

Score	Level	Description
0	No correct assumptions or variables	The group failed to provide any assumptions or variables that could be used to solve the video
1	One assumption or Variable	The group were able to successfully identify at least one variable or assumption that they needed to solve the video
2	Partially Complete set	The group was able to assign several assumptions and variables to the video but did not identify all of them
3	Complete Set	The group successfully provided all of the necessary assumptions and variables needed to solve the video

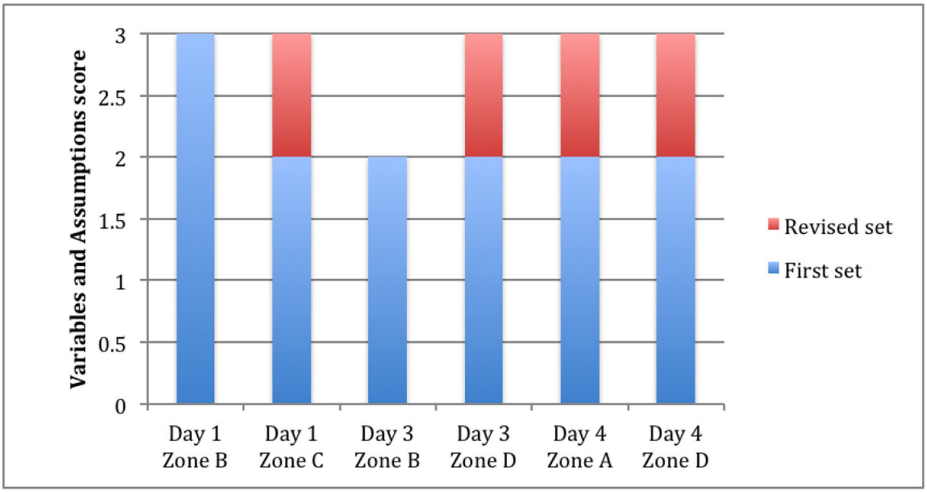

Fig. 7 Variable and Assumption scores for groups before and after the teacher requested the group go over their negotiated set again

Teacher: How did you come up with the initial height?
Student 1: I didn't come up with that, I mean…
Teacher: You made it up?
Students: [laugh]

The teacher then had the group think more deeply about their decision-making process and justifications. When the students asked the teacher to review their work again, he examined their assumptions and variables, and in response to one of their variables, suggested they "have the tank shoot at 90 degrees every time, it isn't really, but it's close". This overall exchange shows that the teacher encouraged students to work out their reasoning themselves, and, in the second case, a slight refinement of their thinking, rather than giving them the answer outright. Combined with the average increase in the group's final completion scores, this seems to indicate that the teacher's orchestration was effective in helping students think deeper, rather than giving them the right answer.

It is worth noting that of Day 1, Zone B, the score was already 3/3 and no additional elements were added which may indicate that the teacher simply asked them to think about it some more, but they did not have to make any changes. On Day 3, Zone B, the group did add another element that was considered significant by the teacher, but they still missed one preventing them from achieving a perfect score. Taken as a whole, the significant changes in groups' completeness scores highlights that the teacher knowing when and where they are needed can have a significant impact on students' knowledge construction and provide important orchestrational support at key moments in students' learning.

Conclusions

This study introduces a new approach to supporting the orchestration of real-time inquiry activities, in which the design of the physical space and the accompanying technologies are carefully considered in parallel with the curricular content. In particular, this study provides evidence for the important role that a real-time teacher tablet, supported by data mining and software agents, can play in reducing the teacher's orchestrational load and supporting him as a wandering facilitator.

The teacher's real-time orchestration tablet became the conduit through which much of the orchestration flowed. By providing detailed information about where students were in the activity at multiple granularities (i.e., small groups and the whole class), the teacher was able to make critical decisions on where he was needed and when the class was ready to progress to the next step in the activity. In particular, the alert that let the teacher know when he needed and the resulting increase in student completion scores, highlights the utility of such a tool to support productive student outcomes. Recent CSCL research has shown how timely teacher support can be critical in student problem solving and idea negotiation. The work of Ingulfsen et al. (2018) and Furberg (2016) showed that students often struggle to make connections between relevant data, and require timely teacher intervention – such as, conceptual support and probing for elaboration – in order to make successful progress. However, without supports to make these moments visible, teachers may miss critical moments, and students may need to compete with peers to get the teacher's attention (Alavi et al. 2009). As such, the design and development of these feedback and visualization tools requires careful consideration. Designers need to understand exactly what teacher needs to see to make better informed decisions, and what elements can be effectively "hidden" to run autonomously.

The ability of this study's orchestration tablet to effectively sort students into groups and place them around the room based on emergent conditions, is an example of how information can be hidden while still supporting classroom orchestration. The teacher did not need to know which students were going to be placed at which spot in the room when he pressed the "sort" button on his tablet (Table 3) – it was enough for him to know that it would be done. Removing this load from the teacher allowed him to focus on the students rather than these managerial tasks. Perhaps the most encouraging feedback on the efficacy these orchestration supports was the teachers comments on the ability of the tablet and agents to reduce his orchestrational load:

> It was such a sort of shifting paradigm kind of lesson, with the pacing and, I don't know, just the kinetics and the motion in the room and kids moving around was a lot to follow, [but] I didn't need to worry about it, it was just taken care of by the various technologies.

Students also noticed the efficacy of S3 in freeing the teacher from many of the managerial tasks in the class, noting that they did not "need the teacher for that any more... he could just focus more on going around and talking to the groups" (student, Jen).

As classroom interventions become increasingly infused with digital technologies to support collaboration and knowledge construction, the real-time state of a student within the class (i.e., their knowledge, interests, or where they are within a particular activity) is increasingly hidden "behind a screen" (Sharples 2013). However, the ability to track the complex connections between students and their peers, the emergent knowledge, and the teacher's goals for learning, offers new support for orchestration that previously would have been too difficult to process manually, especially in real-time. The introduction of real-time software agents can help process this stream of data and connect it to desired learning patterns and teacher needs. Well-designed agents allow researchers, learning designers, and teachers to establish a priori the events they wish the learning environment to respond to, without the explicit need to know *who will fill those conditions* or *when they will fulfill them* prior to the activity's enactment. When done well, these technologies can reduce the teacher's orchestrational load, freeing him or her to do the important tasks of working with students and helping them overcome challenges and refining their thinking.

While this study points to the potential efficacy of an agent-supported tablet system to support classroom orchestration, it admittedly does so in an activity with a fairly linear script. This raises questions about how similar approaches would work in more open-ended scenarios where the script is less-structured (Dillenbourg et al. 2009). Similar to Dillenbourg's (2009) concerns of over-scripting, these kinds of learning spaces may end up over-orchestrating the activity, with students and teachers feeling that the activity is "on rails". The challenge of balancing flexible orchestration while providing the correct level of guidance and regulation is not new (Dillenbourg et al. 2009; Kirschner et al. 2004). As shown by the teacher being able to ask students to revisit work before progressing, we attempted to find this balance. During the exit interview, the teacher noted that it would have been nice for the students to be able to revisit their work if they realized they needed more data from a previous step – a level of flexibility not afforded by this particular classroom script. Based this feedback and our own observations, we have since developed scripts that engage in shorter cycles of discussion and problem solving that allow the class to engage in discussion about next steps and revisit and refine their thinking before going through another cycle (Moher et al. 2015).

A similar challenge concerns how to design orchestration systems that are flexible and robust enough to still function if or when the agents or the system make mistakes (which is likely to happen in any system over time!). Similar to the issues with availability, partition tolerance, and consistency in distributed systems (Kleppmann, 2015), designers of real-time orchestration dashboards will need to consider what happens when issues occur such as dropped data, missed messages, or devices temporarily disconnecting from the system. In the design discussed in this paper, certain orchestration functions could still be conducted by the teacher if the system failed. The teacher could still act as a wandering facilitator going to groups and examining their work, even if he did not receive an alert. However, this would require students to spend time trying to get his attention rather than working (an orchestration challenge similar to Alavi et al. 2009). The teacher could also advance groups to the next step, even if his tablet indicated that not all students were done. On the other hand, problems could arise if students were not put into groups or the content from the database was not properly sent to students' tablets. Designers will need to carefully consider what effect a failure would have on the overall ability of the system to function.

Another possible limitation to this study is that the teacher was well versed in the pedagogical approach, having worked with the research team for several years as a co-designer. Getting teachers acquainted with novel technological and pedagogical approaches is a persistent challenge in CSCL research (e.g., Koh and Hong 2017). However, our goal was to test the capabilities and feasibility of our design, rather than aim for broad applicability. As such, working with an experienced co-design teacher allowed us to focus on the design and implementation. As part of the co-design team, the teacher was well acquainted with how the script was expected to unfold. However, prior to running the class activity, he had not seen the tablet in action. As such, he was responding to the tablet for the first time live. His ability to successfully use the tablet to help his orchestrate classroom activities, points to the efficacy of the tablet's design. Part of this stems from the intentional simplicity of its design. Rather than providing the teacher with everything we could from the live data, we only provided him things that were determined to be immediately and timely actionable (e.g., forming groups, checking student work). This responded to earlier challenges we had with previous versions of a real-time teacher tablet (see Tissenbaum and Slotta 2019).

One final limitation of this work is that it only ran for one session for each class. Developing an orchestrational framework that leverages data over longer scales of time and a diversity of activities (compared to the four steps and two sorts in this activity) would likely require more complex tracking of student interactions and trajectories. For example, some intelligent tutoring systems have seen early success with more complex and longitudinal data analytics (Rubio-Fernández et al., 2019). However, we believe there is a place for both of these approaches to coexist. In many cases, there is limited or non-complex data available for sorting. In the case of the study we present here, the teacher wanted a very specific kind of sort - having the students work with peers they had not worked with before in a modified jigsaw. This would not require the level of complexity, and increased variability, of approaches like *k*-means clustering. Educational designers need to deeply understand and consider these kinds of trade-offs when designing real-time agent-based orchestration tools.

It is worth noting here that smart classroom setups such as these are generally rare. They require a significant commitment to a physical-technical architecture that is at odds with many traditional classroom configurations. However, many, if not all, of these approaches can be achieved in similar lower-cost ways. Active Learning Classrooms (Dori and Belcher 2005) approach classroom design with similar clusters of students working around large, often interactive, shared displays (Charles and Whittaker 2015). While these have been primarily situated in post-secondary classrooms, we are seeing growing adoption of them in K-12 settings (Hod et al. 2016). We are also seeing carts of tablets becoming more common throughout K-12 schools, opening up the opportunity for increased mobility of both teachers and students. What is important from this work is less the particular technologies used, but the kinds of learning, collaboration, and orchestration it supports. Just as the early work with Palm Pilots (Roschelle and Pea 2002) and multi-user computer screens (Szewkis et al. 2011), provided important evidence for future research and classroom implementations, our work work aims to provide a set of generalizable exemplars grounded within the learning sciences for the future research of others.

Understanding the potentially powerful role that agents can play in reducing the time-consuming tasks of sorting students into groups, and providing them timely and context-sensitive materials is something that we believe can have a lasting impact on classrooms broadly. By taking these administrative tasks out of teachers' hands and automating them, we can free the teacher up to spend more time with students, providing more time for classroom learning and collaboration (instead of waiting around for the teacher to make groups and distribute materials manually). In our own design, the teacher noted that the lesson seemed to "gain time" as it progressed, allowing more learning and collaboration to be packed into the class period than he normally expected (Tissenbaum and Slotta 2019). Another key element of this design that we feel can be generalized to other contexts, is to understand what information can help teachers make real-time decisions quickly, and what information might simply increase the teacher's orchestrational load and would be better left for post hoc reflection. In our design, there was a lot of processing going on "under the hood", and yet, we kept the design simple – a limited set of alerts letting the teacher know where students were in the activity and where and when he was needed. This allowed the teacher to keep a heads-up view of the class and did not require him to make complicated assessments of the whole class' learning. This complements the work by Schwarz et al. (2018), which showed how a clear and uncluttered real-time display of small group work can help a teacher intervene at *critical* *moments* in students' problem solving. Their simplified alerts (changing the color of the boarders around each group's work to indicate a specific state), allowed the teacher assess where they most needed in the moment with minimal additional orchestrational load. While this may be valuable in some cases, designers need to carefully assess the load this places on teachers and the resulting trade-offs.

Our design study showed the potential for a teacher tablet that leverages the emergent real-time data in a classroom to help offload much of the monitoring and management tasks to the underlying system. We feel there is considerable potential for technology approaches that free teachers to focus more on the students and act as informed wandering facilitators. The underlying S3 agent architecture played a key role in our work, monitoring student interactions at the individual, small group, and whole class levels. This collection of loosely coupled software agents provides a pedagogically driven blueprint that others can follow within their own CSCL designs. Rather than developing large monolithic monitoring tools, more flexible agents such as the ones in this study, offer the potential for designs that approach orchestration as an ecology, in which agents can work in concert or individually, responding to emergent classroom patterns. As mentioned above, as new tools are developed to harness the huge amounts of data generated in CSCL environments, researchers will need to make decisions concerning their orchestrational flexibility. Key questions moving forward will include understanding what is gained, and critically, what is lost, when we automate some class activities, thus reducing the ability of the teacher to orchestrate elements the class on their own. Similarly, we will need to deeply consider how much is too much data. This work aimed to find a reasonable balance between giving the teacher a lens into the class, while hiding other potentially distracting information away. Moving forward, designers will need to carefully consider how information provided to teachers will be actionable, and more importantly, what the learning outcomes of these teacher interventions will be.

References

Abowd, G. D., & Mynatt, E. D. (2004). Designing for the human experience in smart environments. *Smart environments: technologies, protocols, and applications, 2*, 167–207.

Alavi, H. S., Dillenbourg, P., & Kaplan, F. (2009). Distributed awareness for class orchestration. In . Sage, Distributed Awareness for Class Orchestration.

Anderson, J. R., Corbett, A. T., Koedinger, K. R., & Pelletier, R. (1995). Cognitive tutors: Lessons learned. The journal of the learning sciences, 4(2), 167-207.

Aronson, E. (1978). The jigsaw classroom. Sage.

Baker, R. S., Corbett, A. T., Koedinger, K. R., & Wagner, A. Z. (2004, April). Off-task behavior in the cognitive tutor classroom: When students game the system. In Proceedings of the SIGCHI conference on Human factors in computing systems.

Barab, S. & Squire, K. (2004). Design-based research: Putting a stake in the ground. The journal of the learning sciences, (October 2013), 37-41.

Bielaczyc, K., Collins, A., O'Donnell, A. M., Hmelo-Silver, C. E., & Erkens, G. (2006). Fostering knowledge-creating communities. *Collaborative learning, reasoning, and technology*, 37–60.

Brown, A. L., Ellery, S., & Campione, J. C. (1998). Creating zones of proximal development electronically. In J. G. Greeno & S. V. Goldman (Eds.), *Thinking practices in mathematics and science learning* (pp. 341–368). Mahwah, NJ: Lawrence Erlbaum.

Brusilovsky, P. (2001). Adaptive hypermedia. User modeling and user-adapted interaction, 11(1-2), 87-110.

Charles, E. S., & Whittaker, C. (2015). Active learning spaces: Blending technology and orchestration. In *Exploring the material conditions of learning: The CSCL conference* (Vol. 1, pp. 225-226).

Clark-Wilson, A. (2010). Emergent pedagogies and the changing role of the teacher in the TI-Nspire navigator-networked mathematics classroom. *ZDM, 42*(7), 747–761.

Cole, M. (1996). *Cultural psychology. A once and future discipline*. Cambridge, MA: The Belknap Press of Harvard University Press.

Cook, D. J., & Das, S. K. (2007). How smart are our environments? An updated look at the state of the art. *Pervasive and Mobile Computing, 3*(2), 53–73.

Crook, C. (1998). Children as computer users: The case of collaborative learning. *Computers & Education, 30*(3 & 4), 237–247.

De Winter, J. C. (2013). Using the Student's t-test with extremely small sample sizes. *Practical Assessment, Research & Evaluation, 18*(10).

Design-Based Research Collective. (2003). Design-based research: An emerging paradigm for educational inquiry. Educational Researcher, 5–8.

Dillenbourg, P. (2002). Over-scripting CSCL: The risks of blending collaborative learning with instructional design. Three worlds of CSCL. Can we support CSCL?, 61-91.

Dillenbourg, P. (2012). Design for classroom orchestration, position paper. In P. Dillenbourg, Y. Dimitriadis, M. Nussbaum, J. Roschelle, C. K. Looi & J. Asensio (Eds.), Design for classroom orchestration. Computers & Education.

Dillenbourg, P., Jarvela, S., & Fischer, F. (2009). The evolution of research on computer- supported collaborative learning. In N. Balacheff, S. Ludvigsen, T. Jong, A. Lazonder, & S. Barnes (Eds.), *Technology-enhanced learning* (pp. 3–19). Dordrecht: Springer Netherlands.

Dillenbourg, P., Zufferey, G., Alavi, H. S., Jermann, P., Do, L. H. S., Bonnard, Q., ... & Kaplan, F. (2011). Classroom orchestration: The third circle of usability. In Connecting Computer-Supported Collaborative Learning to Policy and Practice: CSCL2011 Conference Proceedings. Volume I—Long Papers (Vol. 1, No. CONF, pp. 510-517). International Society of the Learning Sciences.

Dillenbourg, P., & Jermann, P. (2007). Designing integrative scripts. In scripting computer-supported collaborative learning (pp. 275-301). Springer US.

Dimitriadis, Y. (2012). Supporting teachers in orchestrating CSCL classrooms. Research on E- learning and ICT in education, (September), 33-40.

Dori, Y. J., & Belcher, J. (2005). How does technology-enabled active learning affect undergraduate students' understanding of electromagnetism concepts? *The Journal of the Learning Sciences, 14*(2), 243–279.

Dunleavy, M., Dede, C., & Mitchell, R. (2009). Affordances and limitations of immersive participatory augmented reality simulations for teaching and learning. *Journal of Science Education and Technology, 18*(1), 7–22.

Fischer, F., Slotta, J., Dillenbourg, P., Tchounikine, P., Kollar, I., & Wecker, C. (2013). Scripting and orchestration: Recent theoretical advances. In *Proceedings of the international conference of computer-supported collaborative learning (CSCL2013)* (pp. 564–571). WI: Madison.

Furberg, A. (2016). Teacher support in computer-supported lab work: Bridging the gap between lab experiments and students' conceptual understanding. *International Journal of Computer-Supported Collaborative Learning, 11*(1), 89–113.

Greene, J. C. (2006). Toward a methodology of mixed methods social inquiry. *Research in the Schools, 13*(1), 93–98.

Hod, Y., Charles, E. S., Acosta, A., Ben-Zvi, D., Chen, M. H., Choi, K., et al. (2016). *Future learning spaces for learning communities: New directions and conceptual frameworks*. Singapore: International Society of the Learning Sciences.

Hollan, J., Hutchins, E., & Kirsh, D. (2000). Distributed cognition: Toward a new foundation for human-computer interaction research. *ACM Transactions on Computer-Human Interaction (TOCHI), 7*(2), 174–196.

Hmelo-Silver, C. E. (2000). Knowledge recycling: Crisscrossing the landscape of educational psychology in a problem-based learning course for preservice teachers. *Journal on Excellence in College Teaching, 11*(2), 41–56.

Hmelo-Silver, C. E. (2004). Problem-based learning: What and how do students learn? *Educational Psychology Review, 16*(3), 235–266.

Ingulfsen, L., Furberg, A., & Strømme, T. A. (2018). Students' engagement with real-time graphs in CSCL settings: Scrutinizing the role of teacher support. *International Journal of Computer-Supported Collaborative Learning, 13*(4), 365–390.

Ishii, H., Wisneski, C., Brave, S., Dahley, A., Gorbet, M., Ullmer, B., et al. (1998). ambientROOM : Integrating with architectural ambient space media. *Chi, 1998*(April), 173–174.

Jennings, N. R., & Wooldridge, M. (1998). Applications of intelligent agents. InAgent technology (pp. 3-28). Springer Berlin Heidelberg.

Jennings, N. (2001). An agent-based approach for building complex software systems. *Communications of the ACM, 44*(4), 35–41.

Jermann, P., Soller, A., & Muehlenbrock, M. (2001). From mirroring to guiding: A review of the state of art technology for supporting collaborative learning. In European conference on computer-supported collaborative learning EuroCSCL-2001 (pp. 324-331).
Johnson, R. B., Onwuegbuzie, A. J., & Turner, L. A. (2007). Toward a definition of mixed methods research. Journal of mixed methods research, 1(2), 112-133.
Kirschner, P., Strijbos, J., Kreijns, K., & Beers, P. (2004). Designing electronic collaborative learning environments. *Educational Technology Research & Development, 52*(3), 47–66.
Kling, R., & Courtright, C. (2003). Group behavior and learning in electronic forums: A sociotechnical approach. *The Information Society, 19*(3), 221–235.
Koh, E., & Hong, H. (2017). Developing professional competency in a CSCL environment for teamwork: Two TPACK case studies of teachers as co-designers.
Kollar, I., Hämäläinen, R., Evans, M., De Wever, B., & Perrotta, C. (2011). Orchestrating CSCL–more than a metaphor. In connecting computer-supported collaborative learning to policy and practice: CSCL2011 conference proceedings (Vol. 2, pp. 946-947).
Lemke, J. L. (2000). Across the scales of time: Artifacts, activities, and meanings in ecosocial systems. *Mind, Culture and Activity, 7*(4), 273–290.
Lui, M., & Slotta, J. D. (2014). Immersive simulations for smart classrooms: Exploring evolutionary concepts in secondary science. *Technology, Pedagogy and Education, 23*(1), 57–80.
Mason, J. (2006). Mixing methods in a qualitatively driven way. *Qualitative Research, 6*(1), 9–25.
Moher, T., Slotta, J. D., Acosta, A., Cober, R., Dasgupta, C., Fong, C., ... & Peppler, K. (2015). Knowledge construction in the instrumented classroom: Supporting student investigations of their physical learning environment. International Society of the Learning Sciences, Inc.[ISLS].
Nussbaum, M., Alvarez, C., Mcfarlane, A., Gomez, F., Claro, S., & Radovic, D. (2009). Technology as small group face-to-face collaborative scaffolding. *Computers & Education, 52*(1), 147–153.
Ontario Ministry of Education (2008). The Ontario Curriculum Grades 11 and 12.
O'Donnell, A. M., & Dansereau, D. F. (1992). Scripted cooperation in student dyads: A method for analyzing and enhancing academic learning and performance. In R. Hertz-Lazarowitz & N. miller (Eds.), interaction in cooperative groups the theoretical anatomy of group learning (pp. 120-141) Cambridge University press.
O'Driscoll, C., Mithileash, M., Mtenzi, F., & Wu, B. (2008). Deploying a context aware smart classroom. Education and Development Conference.
Palincsar, A. S., & Herrenkohl, L. R. (2002). Designing collaborative learning contexts. *Theory Into Practice, 41*(1), 26–32.
Papazoglou, M. (2001). Agent-oriented technology in support of e-business. *Communications of the ACM, 44*(4), 71–77.
Pea, R. D., & Maldonado, H. (2006). WILD for learning: Interacting through new computing devices anytime, anywhere. *The Cambridge handbook of the learning sciences*, 852–886.
Penuel, W. R., Roschelle, J., & Shechtman, N. (2007). Designing formative assessment software with teachers: An analysis of the co-design process. 10.1007/s11412-019-09306-1 Research and practice in technology enhanced learning, 2(01), 51-74.
Purba, S. W. D., & Hwang, W. Y. (2017). Investigation of learning behaviors and achievement of vocational high school students using an ubiquitous physics tablet PC app. *Journal of Science Education and Technology, 26*(3), 322–331.
Roschelle, J., Dimitriadis, Y., & Hoppe, U. (2013). Classroom orchestration: Synthesis. *Computers & Education, 69*, 523–526.
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*(1), 145–168.
Roschelle, J., Penuel, W. R., & Shechtman, N. (2006). Co-design of innovations with teachers: Definition and dynamics. In *Proceedings of the 7th international conference on learning sciences* (pp. 606–612).
Roschelle, J., Rafanan, K., & Estrella, G. (2010). From handheld collaborative tool to effective classroom module: Embedding CSCL in a broader design framework. *Computers & Education, 55*(3), 1018–1026.
Rubio-Fernández, A., Muñoz-Merino, P. J., & Kloos, C. D. (2019, June). Analyzing the group formation process in intelligent tutoring systems. In *International Conference on Intelligent Tutoring Systems* (pp. 34-39). Springer, Cham.
Serenko, A., & Detlor, B. (2002). Agent toolkits: A general overview of the market and an assessment of instructor satisfaction with utilizing toolkits in the classroom.
Sharples, M. (2013). Shared orchestration within and beyond the classroom. *Computers and Education, 69*, 504–506.
Slotta, J. D., & Linn, M. C. (2009). WISE science: Web-based inquiry in the classroom Teachers College press.
Slotta, J. D., & Najafi, H. (2013). *Supporting collaborative knowledge construction with web 2.0 technologies* (In *Emerging technologies for the classroom* (pp. 93–112)). New York, NY: Springer.

Slotta, J., & Peters, V. (2008, June). A blended model for knowledge communities: Embedding scaffolded inquiry. In *Proceedings of the 8th international conference on International conference for the learning sciences-Volume 2* (pp. 343-350). International society of the learning sciences.

Slotta, J. D., Quintana, R. M., & Moher, T. (2018). Collective inquiry in communities of learners. In *International handbook of the learning sciences* (pp. 308-317). Routledge.

Soller, A. (2001). Supporting social interaction in an intelligent collaborative learning system. *International Journal of Artificial Intelligence in Education (IJAIED), 12*, 40–62.

Stanley, K. O., Bryant, B. D., & Miikkulainen, R. (2005). Real-time neuroevolution in the NERO video game. *Evolutionary Computation, IEEE Transactions on, 9*(6), 653–668.

Schwarz, B. B., Prusak, N., Swidan, O., Livny, A., Gal, K., & Segal, A. (2018). Orchestrating the emergence of conceptual learning: A case study in a geometry class. *International Journal of Computer-Supported Collaborative Learning, 13*(2), 189–211.

Szewkis, E., Nussbaum, M., Rosen, T., Abalos, J., Denardin, F., Caballero, D., Tagle, A., & Alcoholado, C. (2011). Collaboration within large groups in the classroom. *International Journal of Computer-Supported Collaborative Learning, 6*(4), 561–575.

Tchounikine, P. (2013). Clarifying design for orchestration: Orchestration and orchestrable technology, scripting and conducting. *Computers & Education, 69*, 500–503.

Tchounikine, P. (2016). Contribution to a theory of CSCL scripts: Taking into account the appropriation of scripts by learners. *International Journal of Computer-Supported Collaborative Learning, 11*(3), 349–369.

Tissenbaum, M., Lui, M., & Slotta, J. D. (2012). Co-designing collaborative smart classroom curriculum for secondary school science. *Journal of Universal Computer Science., 18*(3), 327–352.

Tissenbaum & Slotta (2015), Scripting and orchestration of learning across contexts: A role for intelligent agents and data mining. In Milrad, Wong & Specht (eds.) Seamless learning in the age of connectivity. Springer.

Tissenbaum, M., & Slotta, J. D. (2019). Developing a smart classroom infrastructure to support real-time student collaboration and inquiry: A 4-year design study. *Instructional Science*, 1–40.

van Aalst, J., & Chan, C. K. (2007). Student-directed assessment of knowledge building using electronic portfolios. *The Journal of the Learning Sciences, 16*(2), 175–220.

Weiser, M., & Brown, J. S. (1996). Designing calm technology. *PowerGrid Journal, 1*(1), 75–85.

White, T. (2018). Connecting levels of activity with classroom network technology. *International Journal of Computer-Supported Collaborative Learning, 13*(1), 93–122.

Wooldridge, M., & Jennings, N. R. (1995). Intelligent agents: Theory and practice. *The Knowledge Engineering Review, 10*(02), 115–152.

Yau, S. S., Gupta, S. K., Karim, F., Ahamed, S. I., Wang, Y., & Wang, B. (2003). Smart classroom: Enhancing collaborative learning using pervasive computing technology. In ASEE 2003 Annual Conference and Exposition (pp. 13633–13642).

Affiliations

Mike Tissenbaum[1] · Jim Slotta[2]

[1] College of Education, University of Illinois Urbana-Champaign, Education Bldg, 1310 S 6th St, Champaign, IL 61820, USA

[2] Ontario Institute for Studies in Education, University of Toronto, Toronto, ON, Canada