Multi-touch Technology in a Higher-Education Classroom: Lessons In-The-Wild

Roberto Martínez-Maldonado, Andrew Clayphan, Christopher Ackad and Judy Kay

Faculty of Education and Social Work                 School of Information Technologies
The University of Sydney, 2006, Australia
roberto@it.usyd.edu.au, {andrew.clayphan, christopher.ackad, judy.kay}@sydney.edu.au

ABSTRACT
Inspired by the promise of tabletops for collaborative learning, and building on the many tabletop lab studies, and a few in-the-wild tabletop classrooms, we designed the first semester-long use of a multi-tabletop classroom for two university subjects, with 105 and 40 students respectively. Surprisingly, we found that with just three applications, designed to meet emerging teaching goals, we could support diverse classroom activities. Our technology also featured key minimalist functions that proved effective in enhancing the teacher’s management of the class. This points to a research agenda for the applications and functionalities needed to make tabletop classrooms a reality. This paper describes the design process we followed to deploy multi-touch technology as a classroom ecology and the lessons learnt from the semester-long use in two authentic university courses.

Author Keywords
Surface Computing, Tabletops, Classrooms, Computer Supported Collaborative Learning

ACM Classification Keywords
H5.3. Information interfaces and presentation (e.g., HCI): Group and Organization Interfaces.

INTRODUCTION AND RELATED WORK
The value of collaboration for learning and creative problem solving is supported by a large body of research (Dillenbourg et al., 2009; Stahl, 2006). Interactive tabletops offer potentially valuable affordances to facilitate collaborative classroom learning (Kharrufa et al., 2013b). Key among these are: i) support for group collaboration, because they allow direct face-to-face interaction (Benko et al., 2009; Müller-Tomfelde and Fjeld, 2010) at a large, horizontal display that creates a shared interaction space (Benko et al., 2009); and ii) the potential to exploit the learner’s digital footprints usefully (Müller-Tomfelde and Fjeld, 2010). However, an identified serious limitation is the lack of standard applications and strategies to make multi-touch technology effective in a classroom ecology.

There is a substantial body of work on many aspects of tabletop interaction (see, for example, reviews in Benko et al., 2009; Müller-Tomfelde and Fjeld, 2010; Scott et al., 2003). A growing slice of this work tackles the particular demands of learning contexts (Dillenbourg and Evans, 2011; Evans and Rick, 2014; Higgins et al., 2011). Yet, there has been little work involving actual authentic classroom deployments. As Kharrufa et al. (2013a) highlights, we are in urgent need of “studies in realistic environments” for important progress. This is because of the complex interplay between learning designers, class teachers and students, as well as other important pragmatics, such as the implications of repeated use of applications and the changes due to long term experiences of the teachers and students.

Just four key multi-table classroom studies have been reported, each varying in-the-wild. The first tabletop classroom, SynergyNet (AlAgha et al., 2010), was a special studio with four student tabletops and one with special controls for the teacher. Through a series of studies, researchers analysed how elementary school students collaborated at the tabletops and how a teacher adapted their pedagogical strategies (Higgins et al., 2011). A second study used tangible devices called TinkerLamps (Do-Lenh et al., 2012). This tackled the design of orchestration tools, helping teachers manage the flow of class activities for an apprenticeship program. MTClassroom (Martinez-Maldonado et al., 2013b) was the first that reported multi-tabletop technologies used in the classroom as part of actual authentic curricular activities. It explored the impact of providing the university teacher with real time visualisations to support awareness of student work and progress. Kharrufa et al. (2013a) also reported authentic sessions for elementary school students, using this to empirically discover the type of functionalities that are preferred in their tabletop classroom deployments. The authors of the last two studies combined their experiences and concepts of distributed cognition to create a comprehensive set of design recommendations for tabletop classrooms (Kharrufa et al., 2013b) which we build upon.

Figure 1. Our multi-touch classroom ecology – top view.
Inspired by the promise of such work, we designed a multi-touch technology classroom ecology for semester long use in support of the practical work of university subjects (see Figure 1). One was an undergraduate HCI subject. The second was a postgraduate Pervasive Computing subject. Both made extensive use of group work in two very different ways: (1) for the learning activities run in each lab class; and (2) for a major group project that spanned the entire semester. This is typical of many tertiary subjects, and contributes to developing key generic skills in communication and group work.

The contribution of this paper is two-fold: i) the description of the design process that we followed to successfully deploy interconnected multi-touch tabletops as a classroom ecology and ii) the documented experience and lessons learnt from the semester-long use of interactive tabletops in two authentic university courses.

The next section describes the design of our technological infrastructure, drawing on the guidelines for designing tabletop classrooms (Kharrufa et al., 2013b). Then, we present the teacher’s learning design for each week during the semester. We then describe our three applications and critically review what we learnt from their use. We conclude with a research agenda for building core applications and functionalities for tabletop classrooms.

**DESIGN OF THE MULTI-TOUCH CLASSROOM ECOLOGY**

The design of our multi-touch technology classroom ecology has been the result of an iterative process of enhancements driven by authentic learning and pedagogical requirements. Our current work builds upon previous studies conducted in two separate sessions in the classroom in previous years (Martinez-Maldonado et al., 2012). The first study (see Table 1, column 1), was conducted for one week (week 7 of the semester) in a university course. This focused on the integration of the multi-touch technology and sensors to automatically capture traces of student’s data and class activity to be used by the teacher for after-class reflection (Martinez-Maldonado et al., 2012). Then, a year later, a second study (in week 6 of the semester) focused on providing the teacher with access to the captured data during the class as well as basic tools to control the classroom technology (Table 1, column 2) (Martinez-Maldonado et al., 2013b). These two studies involved large classes split into tutorial sessions of 20-25 students for two different tertiary courses on *Business* taught by the same teacher. The teacher conducted similar learning activities in both courses based on concept maps. Consequently, the results of these two previous studies could not be easily accounted to the technology itself, the skills of the only participant teacher, the software application or the area of study.

This paper presents, for first time, the documented experience of continued usage of our multi-touch technology in the classroom for several weeks featuring a more diverse variety of teaching styles (4 teachers), learning applications (3 applications), and course lessons (a different weekly activity in each of the two subjects).

Our design is also based on recent guidelines (Kharrufa et al., 2013b) which are based upon theories that have emerged from decades of research on CSCL (Computer Supported Collaborative Learning). They are grounded in theories of distributed cognition and more recent work on classroom orchestration (Martinez-Maldonado et al., 2013a). The guidelines are:

- **G1.** Support visual and auditory interactions across planes of individual, group and class work.
- **G2.** Support movement of learning material between groups and to class, using a rich digital ecosystem as appropriate.
- **G3.** Support transitions across planes so that teachers can bring the class together and can share one group’s work with the class.
- **G4.** Provide accountability based on user identification and tracking of student and teacher activity.
- **G5.** Consider ownership and access rights.
- **G6.** Provide a private teacher space with information supporting their awareness of group processes and progress.
- **G7.** Support for staging and scripting of the learning activities.
- **G8.** Capture the processes as well as final results of learning activities.
- **G9.** Extend the impact of the task beyond the tabletop session duration so that teachers can prime the learning tasks with resources, and output constructed available for learners after the class.

<table>
<thead>
<tr>
<th>1 week in semester 2012</th>
<th>1 week in semester 2013</th>
<th>Full semester</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data capture</td>
<td>Real-time visualisations and teacher’s tools</td>
<td>Long-term usage and support beyond the classroom</td>
</tr>
<tr>
<td>1 teacher</td>
<td>1 teacher</td>
<td>4 teachers</td>
</tr>
<tr>
<td>1 subject</td>
<td>1 subject</td>
<td>2 subjects</td>
</tr>
<tr>
<td>1 application</td>
<td>1 application</td>
<td>3 applications</td>
</tr>
<tr>
<td>230 students</td>
<td>140 students</td>
<td>145 students</td>
</tr>
<tr>
<td>Business</td>
<td>Business</td>
<td>Engineering</td>
</tr>
</tbody>
</table>

*Table 1. Design process across three years of research.*

![Figure 2. Our multi-touch classroom ecology – side view.](image1.png)
defined the space shown in Figure 1 and Figure 2. The layout ensured guideline G1 was supported, since all tables were in easy earshot and line of sight from the teacher’s perspective. Its five tabletops each comfortably seated 5 students, with just enough space for teachers to walk around. Two wall mounted plasma TV displays, one projector and a standard whiteboard supported whole class sharing of the learning artefacts constructed at each individual tabletop (G2-3). These displays could also be used by the teacher to present regular slides.

We used tabletops that could identify user touches (G4) using an overhead depth sensor (Martinez-Maldonado et al., 2011), hence providing foundations for accountability of student’s actions. A tablet for the teacher provided an awareness dashboard (G6) and control of the digital ecology (G3,7) (Martinez-Maldonado et al., 2013b; Martinez-Maldonado et al., 2012).

The classroom digital ecosystem and infrastructure, shown in Figure 3, provided two critical stores: the orchestration server and the external student project server. The orchestration server mainly provided services within the classroom. It was used to log and keep real time data about student’s activities, which could be used to visually present aspects of the learning process and student’s participation in the teacher’s awareness dashboard (G6, 8). This server also maintained the class lists and details of each student’s project team.

The student project server, enabled each table to access a Trac1 repository for the group using it during class time (G9). This provided students with online access to their own materials linked with the semester project, both at their tabletops and from other computers, in and outside of the classroom. It also ensured that the group controlled their own persistent data store, keeping their own data private (G5). Importantly, it provided persistence of the artefacts that students created in each class activity across sessions during the whole semester. The learning applications we used in the classroom were either directly connected to the Trac server to allow changes in real time, or artefacts were uploaded immediately following the learning activities after-class. This key function has not been reported in previous tabletop classrooms.

STUDY

Our multi-touch technology classroom ecology was designed for use for the majority of the weekly 1-hour tutorial classes for two subjects. The undergraduate HCI subject had 105 students, with 6 tutorial classes, taught by 3 teachers (tutors). Pervasive Computing was for postgraduate coursework students; with 40 students, in 2 tutorial classes taught by 2 tutors. In both classes, the tutorial was devoted to either: exercises to consolidate and complement lecture material; or time for groups to work on their major project. Importantly, class activities involved small group discussions and tasks. There were two levels of group work: collaboration on small learning activities; and long term work on the group project.

Classes started in the multi-touch classroom in Week 5; Pervasive Computing students did Android programming in a different room until then and did demos of them in Weeks 7 and 12. For HCI classes, tutorials were in a conventional lab classroom until Week 5 which was the time that the project groups were first formed. In the planning of the semester’s learning activities, the main teacher realised that the usual small group tutorial tasks could be readily mapped onto just three applications which could easily be integrated into our multi-touch classroom ecology to serve the diverse learning objectives for the semester. We now briefly describe the design of our learning applications and the functionalities of our technological ecology that meet the classroom design guidelines, summarised in Table 2.

The learning applications are:

- B: ScriptStorm, a brainstorming creativity application that explicitly supports scripting (G7) (Brown et al., 2013; Clayphan et al., 2011) and which has a reflection

---

1 http://trac.edgewall.org
phase that we used to help students self-grade their work. We designed learning activities using it for brainstorming ideas for target users, their goals and needs for the group project (Week 5 HCI and Pervasive), defining user tasks (Week 6 Pervasive, Week 7 HCI) and designing prototype evaluations (Week 10 Pervasive).

- C: CMate, a concept-mapping application designed for classroom orchestration (Martinez-Maldonado et al., 2012) (G2,3) in a classroom of tabletops that are augmented for user identification (Martinez-Maldonado et al., 2011) providing the basis for individual accountability (G4). Concept mapping is a well-established technique that promotes meaningful learning and is effective in problem solving activities (Stahl, 2006). We used CMate for revision activities in Weeks 7 and 13 for HCI, and Weeks 9 and 13 for Pervasive.

- WG and WO: Well-Met, a meeting application designed to support long term group projects. It provided an agenda tool with visual support to help groups work through items in a meeting (G7). At the core of its design was persistence of the meeting decisions and access to the groups’ digital online artefacts (G9). We distinguished two different ways to use Well-Met; WG (where the agenda was freely defined by group members), used for student group meetings in Weeks 6 and 11 of HCI and Weeks 6 to 11 of Pervasive; and WO (where each group’s agenda was defined by the teacher), used for orchestrating a meeting, linked to relevant resources (Weeks 9 and 10 in HCI).

<table>
<thead>
<tr>
<th>Wk</th>
<th>HCI</th>
<th>Pervasive Computing</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>User tasks &amp; goals (B)</td>
<td>System goals (B)</td>
</tr>
<tr>
<td>6</td>
<td>Prototype sharing (WG)</td>
<td>Think-aloud (B) &amp; Evaluation planning (WG)</td>
</tr>
<tr>
<td>7</td>
<td>Think-aloud (B) &amp; Mid-term revision (C)</td>
<td>Demo prototypes (-)</td>
</tr>
<tr>
<td>8</td>
<td>Demo prototypes (-)</td>
<td>Planning (WG)</td>
</tr>
<tr>
<td>9</td>
<td>Cognitive walkthrough (WO)</td>
<td>Mid-term revision (C) &amp; Planning (WG)</td>
</tr>
<tr>
<td>10</td>
<td>GOMS (WO)</td>
<td>Tasks for final demo (B) &amp; Planning (WG)</td>
</tr>
<tr>
<td>11</td>
<td>Refine presentations (WG)</td>
<td>Group Meeting (WG)</td>
</tr>
<tr>
<td>12</td>
<td>Project presentations (-)</td>
<td>Project presentations (-)</td>
</tr>
<tr>
<td>13</td>
<td>Course revision (C)</td>
<td>Course revision (C)</td>
</tr>
</tbody>
</table>

Table 2. Learning activities during the semester (Tabletop application used: B-Brainstorming; C-Concept Mapping; Well-Met WG- Group Meeting; WO-Orchestration).

With three learning applications, each used at least twice during the semester in each course, students needed to learn to use multiple new interfaces, all within tight class time constraints. For this, the first two applications offered minimalist functionality so they could be learnt by students within minutes on first use. The third application provided a larger range of options, but functions resembled those already found in the group’s online project workspace, thus minimising learning difficulties – as all students were already familiar with the main concepts and objects that could be accessed. The main teacher reserved time for the sessions when a new application was used for the first time.

**ANALYSIS OF THE MULTI-TOUCH ECOLOGY**

We draw upon diverse sources of evidence to analyse the applications and functionalities provided during the semester. In each class, at least one member of the research team was present for the 8 class hours of the week. They acted as an observer of the classroom as a whole, the individual groups and the teacher. All applications logged fine-grained activity, both to display information in the teacher’s dashboard and for further analysis. The backend persistent store (Trac) also preserved the main work produced by students. A questionnaire was completed by 89 and 33 students at the end of the semester, for the HCI and Pervasive courses respectively. Finally, semi-structured interviews were conducted with three of the four teachers, once at the middle, and again at the end of the semester.

In this section we describe our three applications in terms of their core collaborative learning support and students’ perspectives. Note that all three applications shared the common infrastructure described above, as it was critical in meeting the general design guidelines. All three were explicitly designed for tabletop use, with careful consideration of known problems. For example, the effect of orientation for readability of text was addressed by a combination of careful layout, duplicated interface elements or functions for changing orientation. For ScriptStorm and Well-Met, where there was a need to provide quick and convenient text input, we provided physical keyboards. To address clutter, the interfaces included trash-can facilities in the corner of the tabletop displays. In addition, each of the applications set up the tables for each group, taking the student’s names from their group website (the Trac project server) and presenting these on each of the tabletop interfaces.

**CMate**

Our concept mapping application is representative of a class of quite generic tools to support group cognition and collaborative knowledge building (Stahl, 2006). A concept map is a directed graph, where nodes are concepts (e.g. “usability-method” or “think-aloud”) and these are linked to create propositions (e.g. think-aloud is a ‘usability-method’). Typical use of concept mapping is to pose a focus question and ask students to create a map that shows their response. Concept mapping helps learners externalise their understanding, and, when done collaboratively, creates opportunities for explanation, argumentation, negotiation, regulation and reflection, all valuable collaborative learning processes (Stahl, 2006).

Our tabletop concept mapping app, CMate (Martinez-Maldonado et al., 2013b) is shown in Figure 4. CMate is minimalist, with just three interface tools: a list of suggested concepts, provided by the teacher; a list of teacher-defined name links, presented when the students connect two concepts; and a virtual keyboard so students...
can create new concept and link names. CMate was the only application used in previous classroom studies and originally designed to support a teacher dashboard, the MTDashboard (Martínez-Maldonado et al., 2013b). This dashboard supports basic classroom management functions: loading an initial scaffolding map; blocking all tabletops for whole-class activities; and sending a selected tabletop concept map to the vertical displays. At the end of each class, each group’s map was loaded to their Trac project site so students could review it.

Teachers used CMate for a total of 4 distinct revision tasks within 16 tutorial sessions. The main teacher primed the map with initial concepts and propositions to help students get started, important for the tight time constraints. CMate also accepts a master map so that the teacher dashboard can show each group’s progress in creating propositions, distinguishing those crucial ones in the teacher’s map. This enables the class teacher to readily track each group and to decide which group, most may need attention.

**ScriptStorm**

Collaborative brainstorming is a widely used creativity tool, intended to help groups quickly broaden their scope of ideas to consider. It normally has two main stages. In the first stage, ‘idea generation’, team members should create ideas, working in parallel, calling them out to help others spark new ideas. All participants should aim to participate by creating new ideas. There should be no criticism or assessment at this stage. The second stage involves collaborative group work to review the ideas, group them and decide which are most promising. As in many group learning activities, it can be valuable to add a reflection stage, where students have an opportunity to assess whether they met the learning goals. This aspect was critical for this application, as it was used at the start of the groups’ projects.

Our classroom implementation required refinements to ScriptStorm (Figure 5), which was initially designed to explore how to enhance tabletop collaboration with the system supporting (or enforcing) scripting. ScriptStorm interfaced with the teacher’s dashboard, similarly allowing: a view of real time statistics on each group; loading each brainstorm; moving between stages; and blocking tabletops. Importantly, we made use of ScriptStorm’s reflection phase to ask groups to answer questions that helped ensure they considered the most critical ideas that the teacher intended them to take into account. Each class concluded with activities for reflection and sharing, making use of the wall displays.

After each tutorial, the assets built by each group, (the ideas, categories and visual relationships), were uploaded to that group’s Trac project space. Similarly to the final collaborative concept maps, this supported later access to the material generated by students during class for their use throughout the semester. In total, ScriptStorm was used for 5 distinct brainstorming activity tasks within 18 tutorial sessions.

**Well-Met**

Well-Met was designed as a tabletop application to support small groups coming together for regular meetings, as part of a long term project. From its earliest design, it was clear that it needed a persistent store that would be useful for individuals to use between meetings. Well-Met was designed with a family of such stores in mind. For the classroom, Well-Met natively implemented an interface for the Trac project server. Its interface was influenced by the functionality afforded by Trac. Well-Met (Figure 7) contained the following elements: notes, that could be taken during the meeting and stored with the record for that meeting; task creation/allocation, allowing groups to record items to be completed, and allocated appropriately to team members; a big picture tool, to provide a quick visual overview of project milestones; and an agenda facility, to define topics to be discussed in preparation for a meeting – every item associated with a time element and optional digital artefacts. Each of these elements was easily accessible through toolbars, a separate toolbar, placed in front of each user.

All actions at the tabletop were automatically synchronised to the web-based Trac instance. Laptops or other personal devices could also be used, to allow for multiple views of the information, and to allow a seamless bridge between tabletop devices installed in the environment, and those carried by people. Well-Met further supported browsing of files uploaded to the

---

2 others include Phabricator (http://phabricator.org/) and Jira/Confluence (https://www.atlassian.com/)
project webpages, for example from a previous meeting or between classes. Furthermore, the agenda tool, also allowed students to mark if items were complete, providing a basic form of time management, both for the students and any teacher (who might have happened to walk by a tabletop). In total, Well-Met was used for meetings in 9 learning activities (4 in HCI, 5 in Pervasive) for a total of 38 tutorial sessions.

**Teacher's Dashboard**
Figure 6 shows an example screenshot of the teacher’s dashboard displayed on a handheld device. This is the interface that the teacher could use to access the orchestration server. It provided three key forms of control over the tabletops (Figure 6, A), the ability to freeze them – gaining the attention of the whole class (guideline G3), the ability to move through the stages of a class script (G7), and finally the ability to send content from specific tables to the vertical displays (G3) (Figure 6, B). Additionally, it provided three key functionalities to support the teacher’s class awareness (G6). First, it presented a visual representation of the class script indicating the learning activities and the order in which the associated applications should be loaded in each table during the class (see runtime script visualisation, Figure 6, C). This was represented with a horizontal progress bar where each activity corresponded to one section of the visualisation. The section was coloured red when the learning activity ran past the planned time so the teacher could decide how to compensate for time loss during the tutorial.

Second, the dashboard presented a visualisation per group/table indicating a specific aspect of group participation or performance during the class (Martinez-Maldonado et al., 2013b). For example, Figure 6-D shows five coloured concentric radars that indicate the distribution of participation of students within each group. Lastly, it showed alarms (e.g. Figure 6, E) when the system detected either teacher-defined misconceptions or when a group was performing well. This was indicated with a red or green rounded square around the relevant group’s visualisation.

**THEMATIC EMPIRICAL ANALYSIS**

**Beyond Novelty Effect**
Discussions from previous studies using tabletops for education have reported the novelty effect of this emerging technology, especially because tabletops were used under lab conditions (Do-Lenh et al., 2012; Higgins et al., 2011) or for a short time (Martinez-Maldonado et al., 2013b). In our study, the multi-touch classroom was the regular physical space where the weekly tutorials
were conducted. Based on the relevant literature on the use of tabletops in the classroom (Kharrufa et al., 2013a; Kharrufa et al., 2013b) and our first-hand experience in previous semesters (Martinez-Maldonado et al., 2013b), the long term usage of tabletops required: multiple applications to conduct different learning tasks, support for the teacher to orchestrate the technology and persistent storage for the work students created in class.

At the end of the semester, students indicated agreement with “I liked the tabletop lab for group work” on a 5-point Likert scale. For 33 students from Pervasive, the median was 4; and for 89 students from HCI, the median was 3. In associated open comments, students indicated that using the tabletops in the classroom was fun, engaging, interesting, cool, new, a nice opportunity, attractive, worthwhile and awesome for 17 from a total of 98 comments. There were 8 positive comments about supporting group-work (helped collaboration, bonding, discussions, share ideas). Finally, only 2 students mentioned the value of the persistent store. This suggests that one of the crucial functionalities that had been missed in all previous multi-tabletop classrooms was already expected by students as a basic feature to support continued activities.

Distributed Orchestration

There is a pedagogic decision, and a tension, between giving students guidance that is not enforced as in the agenda tool from Well-Met, versus tight scripting, controlled by the teacher (either in a pre-set script or the dashboard controller) – such as which the brainstorming (ScriptStorm) and concept mapping (CMate) applications provided. In other words, for applications such as Well-Met, where the orchestration load can be distributed among students, theoretically the teacher does not need to manage the execution of a class script and hence their orchestration load gets reduced. However, we found, in our very time-restricted classes, guidance-alone (when Well-Met was used as a proxy tool for orchestration) put considerable load on the teacher, as each group progressed at different rates; making it difficult for the teacher to track what was happening in the classroom.

This trade-off is affected by the type of activity. For activities that should have a defined structure, such as revision (e.g. as it was the case for the use of CMate) – where the teacher wanted to be certain all students were exposed to important concepts – it is advisable that groups proceed at the same pace and this is where a common learning script can help drive this process. Our infrastructure allows novel mechanisms to be provided, that can aid the teacher in orchestration of the classroom, such as an interactive dashboard with analytics, notifications, and even on-the-fly comparative analysis between groups.

By contrast, for activities that permit themselves to less structure (from the perspective of the teacher), such as meetings – where it makes more sense for each group to be independent and progress at their own pace, visual information from a dashboard while still important, may become less contextually relevant at times. In the sessions with group meetings, teachers voluntarily stopped using the dashboard (sessions 6, 9, 10 and 11 in HCI; and 8 and 11 in Pervasive). By contrast, teachers used the dashboard in all the sessions when a script was designed by the teacher for the whole class (sessions 5, 7, and 13 in HCI; and 5, 6, 9, 10 and 13 in Pervasive).

Awareness of the Class Script

Even though our classroom experience was framed within a realistic and in-the-wild scenario, we defined a series of conditions for some of the weekly tutorials to study the impact of making explicitly visible the enactment of the class script to the teacher. Two main conditions consisted of showing or hiding the runtime script visualisation (see Figure 6, C). We counter-balanced these two conditions among tutorials, teachers and courses. As the teaching load of the teachers was different (teaching 3, 2 or 1 session(s)), the cross-validation was not perfect, but provides a good basis to understand the impact of showing the visualisations. During the post-tutorial interviews with the class teachers, all gave positive feedback about the use of the runtime script visualisation.

Three of the teachers described the information provided by the script visualisation as “very useful”, “easy to understand” and that it “helped them have a better idea about the tasks [they were] supposed to conduct”.

In agreement with the above, upon compiling the actual times teachers spent in each stage of their lesson plan, we found evidence that, when the visualisation of the script was not presented, some tasks took longer or shorter than stated in the class plan (increased variability). For example in Week 7, for an idea generation task, teachers took 8.6 and 13.2 minutes on average, when the script visualisation was visible and hidden respectively. This caused an extension of that activity beyond the time initially allocated for it and a large reduction of the time for other important tasks in the whole class activity. We also found a significant effect of the script visualisations on the matching between each planned task and the enactment (the mean difference between each individual task’s planned duration and their enactment was 36.2% (±34) and 78% (±85) over the planned time respectively when the visualisation was shown and hidden; (Mann-Whitney U-Test, U = 36.5, Z = -3.66, p < 0.01). The positive effect on the enactment of the class script by the teacher was explained by one of the teachers as follows: “the script visualisation specifically helped me figure out when I had to compensate for the duration of certain tasks, for example, if I could see many red sections (overtimed tasks) I knew I had to consider that for the current stage”.

Teacher’s Awareness and Provision of Feedback

A number of visualisations of the groups’ participation and performance (Figure 6, D), and notifications (E) were also displayed on the dashboard. In the case of the notifications, according to our observations of teacher’s movements in the classroom triangulated with the logs, an average of 88% (±19) of the times that teachers got a notification, they immediately attended to the group that triggered the notification to either provide feedback or have a closer look at their work. The process commonly followed by teachers after a notification appeared on the
dashboard was described by one of the teachers like this: “I would tap on the notification and read [more information] about the problem. Then I would move to the table and have a conversation to understand what [students] did and help them if needed”.

For the case of the alerts about students’ misconceptions, we analysed that within two minutes of receiving teacher feedback, in 70% of the cases, the attended group addressed at least one of the relevant misconceptions that triggered the notification. This suggests that our system may have made teachers more aware about students’ misconceptions and guided their attention to groups that required it, to immediately address raised concerns. All the interviewed teachers considered that these notifications were useful. As stated by one of the teachers, “the ‘red notifications’ were useful because I could easily identify if there was a problem in a group and try to fix it, as soon as I saw it”. Another teacher stated that, as a result of their immediate feedback, if there is some problem, “students can immediately take action or [the teacher] can make some suggestions so they can figure out what is the problem and learn something”.

For the case of the positive notifications, even though there were fewer, they proved to be useful to teachers (for 13 of the 18 positive notifications presented in the dashboard). These were used by teachers in a different way: teachers provided positive feedback to students “to encourage groups to stay motivated”. However, two teachers mentioned that these notifications “had the lowest priority but [they] still used [them], to go to the group, [to] let that group know that they were progressing positively” if no other group had unsolved misconceptions to attend to first.

Overall, all the interviewed teachers found that one of the most beneficial functions of conducting the classes in the multi-touch classroom ecology was the awareness support provided by the small group visualisations, the visualisations about the script and the notification of misconceptions or positive group progress. For example, this was described by one teacher as follows: “it was really important that I could see the script visualisation in all conditions of the dashboard, along with the information about the size of student’s solutions... those are the things I wouldn’t be able to see by myself”. Another teacher added: “I can be relatively aware about what students are discussing, but not about the script or the progress of student’s solutions”. Finally, a third teacher added that “the notifications had priority because I wanted to help them fix the problem (the misconception)”. 

Concept mapping, for Knowledge Sharing and Revision
In our multi-touch classroom, concept mapping was used in previous years by the teacher to conduct problem-posing tasks that required collaborative work between learners. The teacher described the support provided by CMate as follows: “[even when] not everyone was interacting with the tabletop all the time, but being around the table helped students discuss their points of view; they were speaking a lot and this is good from a learning perspective”. However, concept mapping is not only a learning tool; another powerful use of concept maps is for evaluation (Novak, 1995). In our classroom, concept mapping helped teachers conduct revision exercises focused on the themes that were more problematic for the students during the written exams. These interventions using concept mapping as a revision tool proved very valuable for students, with students rating these exercises 3.85 (out of 5 on a Likert scale).

Additionally, CMate was designed for orchestration and helping teachers. This meant that the learning design relied on the teacher to use a discussion period at the end of the class to review the concept maps, critiquing them as part of the sharing process. In practice, this was difficult to do within the tight time constraints of the class. Teachers spent on average a quarter or more of the lesson time for this final class sharing and discussion.

Brainstorming, Scaffolding and Reflection
With a combination of idea generation, categorisation, group sharing and individual reflection stages, ScriptStorm provided many ways to construct the tutorial. For example, in Pervasive – Week 6, the topic “brainstorm about tasks you will ask participants to do in your evaluation” related to the on-going group project work. The activity was comprised of: idea generation, idea categorisation, student reflection, and student feedback. Within the idea generation stage, students jointly decided on idea placement and colour. This flexibility of choice was received well, with an average rating of 4.2 (out of 6). For student feedback, brainstorming was found to promote group immersion (4.5/6), aid in the whole-class sharing of knowledge (4.5/6), and help build dialogue between each of the groups and the facilitating teacher, in regards to goal attainment (self-reflection) (4.2/6).

As stated earlier, reflection played a major theme with brainstorming, with all HCI weeks and half of the Pervasive weeks having time dedicated for group self-reflection. Here, students evaluated their outputs, in relation to a set of questions, or exemplar ideas given by the teacher. This allowed students to rate how well they believed they met tutorial goals, and as a way to promote awareness in the class. In practice, we found students were overly generous in their self-assessment of how well they thought they met tutorial goals. This invariably led to input from teachers (as they walked around the room, viewing groups making their choices), and at some points, interjecting in order to help students better reflect on their understanding. We consider that this aspect, the balance between self-assessment, and teacher led discussion is important within the class activity. For the types of learning activities where group work is most valuable, there are often no simple answers that inexperienced learners, even working in groups, can reliably use. This is precisely where a teacher has a key role to play. Additionally, information collected from these self-reporting exercises is beneficial to teachers. It can allow a teacher to better unpack and understand how students and groups truly perceive and understand set tasks and activity objectives.
Small group meetings
The same Well-Met application was used to support meetings, and for orchestration of group learning activities. The differences between the two types of meetings were, one where the group used the agenda function to progress over a set of tasks at their own rate, and the other mode, was where the teacher pre-loaded the agenda with related learning materials for the tutorial. For example, the Extended Cognitive Walkthrough activity in Week 9 of HCI had: ‘a sample interface design document attached’, ‘assumptions to discuss in defining the user’s mental model’ and ‘a set of tasks to use’. Each group recorded their response using the note tool. We found some groups used the agenda as intended, consulting it regularly, keeping to time and marking items as completed as appropriate. We also viewed some groups requiring encouragement from the teacher to do this, before they became more familiar with the process.

Challenges in Classroom Deployments
Having a classroom with multiple connected devices, sensors, a real time dashboard, an orchestration server and a projects’ server, lent itself to some interesting technical, educational and development challenges. The technical challenges included how to allow multiple technologies and applications to co-exist, and how to orchestrate these, given that direct access to the computers running the tabletop devices is not possible during class (since students are seated around the devices – and the computer itself is within an enclosure). To do this, we defined a novel protocol, which each application implemented against, to provide the seamless experience of being able to run multiple applications synchronously.

For learning challenges, the main issue was to find a way to transform the high level description of the tutorials, from the teacher’s design, into items that function within the technologies used – and to keep to the spirit of a lesson plan. To do this, we devised a high level XML description file, which mapped teachers’ actions to stages, tasks, applications and learning resources. While in this classroom, the learning script was built by the research team (using input from the teacher) moving forward, an interface for drag and drop composure of such lesson plans will be required.

For deployment, we predicted that software when used by students needed to be robust and ready; as any such failure of the software we believed would lead to negative opinions being formed, and carried into future tutorials. It was exactly these issues we found in moving from solid lab prototypes to deployment. The most common negative comments at the end of the semester were that 16 students mentioned software failures, 13 referring to unspecified usability problems and 10 on lag problems caused by the capabilities of our hardware affecting productivity. Most were due to the early uses of Well-Met, with its real-time integration with Trac and the dependency on an external server. Others were due to new bugs not seen in lab use, one tabletop developing hardware problems and one having a less ergonomic enclosure. Notably 4 students commented that they played games and were distracted, in line with our observations of some HCI students who seemed very set upon trying to make the system fail. This may be a negative experience of just those aspects that other students described as fun and engaging.

LESSONS LEARNT AND CONCLUSIONS
Inspired by promising work on the benefits of tabletops for future classrooms, we wanted to explore how we could move towards a technology-enhanced classroom, with tools supporting collaborative learning and project group work. Like others who have created authentic tabletop classrooms (Kharrufa et al., 2013a), we consider that it is timely to create such classrooms to gain insights into the nature of potential future classrooms that make use of a rich digital ecosystem that includes interactive tabletops.

We presented the design and authentic full-semester deployment of a digital classroom ecology that integrates multiple interactive tabletops to support small group knowledge building and meetings. In our initial consideration of the learning design, we were rather surprised to find that our small set of applications seemed well suited for supporting a useful range of collaborative learning activities through much of a semester’s regular practical work. We had designed each of these applications with quite different core research drivers. Only the concept mapping application had been explicitly designed for classroom orchestration and subject to previous comprehensive evaluation in-the-wild (Martinez-Maldonado et al., 2013b; Martinez-Maldonado et al., 2012). But that experience, combined with available design guidelines (Kharrufa et al., 2013b) enabled us to perform a principled design of the physical space, the digital ecosystem and infrastructure within the classroom and the persistent student project server. This latter functionality supported continued learning activities beyond the classroom sessions. Future research needs to test these guidelines further, refine them and provide case studies of how to materialise them.

We also showed the operationalisation of an orchestration server that helps teachers manage and coordinate available classroom technology including for example: deciding what to show on the vertical displays; the pace of the class script; and the ability to temporarily block technology interactivity to gain the attention of the students. From our experience, a key finding was that situating a large interactive display as a horizontal surface in a workspace encouraged group members to work around it in a socially cohesive and conducive way. Overall, the main lessons learnt from our research included:

- A number of tabletop applications can be adapted according to a standard so they can be plugged-in and ‘orchestrated’ by classroom technology. This may offer a more varied mix of learning tools that can be used in classrooms, functions for student guidance or for teachers to control transitions between learning activities, tasks and spaces, as we had used in the sessions where two of the three applications - CMate, ScriptStorm or Well-Met - were used in the same session.
From our experience, there are particularly promising applications, such as Well-Met that can become the standard (Benko et al., 2009) for tabletop classrooms, that can support **flexible and distributed orchestration** and encourage student’s self-coordination.

The classroom technology must be linked to an appropriate **persistent store of learning products**. For the future, the nature of that store may be a Learning Management System (LMS), an ePortfolio, a cloud service or a project support system like Trac. The purpose is to provide continued support through linking activities inside and outside the classroom. There is no current clear choice and this deserves exploration and research.

The design and deployment of these types of classroom ecologies need **primitives to make it easy for teachers to create the activity plan** and the design of the tasks in each class. In our experience, the research team helped the main teacher translate her activity designs into an XML script that could be enacted by the classroom technology. Tools to aid the construction of the teacher’s learning designs may also be included so teachers can create and run their own design in the classroom without depending on intensive technical support.

The interactive tabletops should support **user identification** (Martinez-Maldonado et al., 2011) both for creating effective interaction with the application and for ensuring accountability, particularly for students. User identification also makes it feasible to provide awareness tools that show progress and processes in the learning activities at runtime, especially for teachers in the classroom, but also for students when there is time available for reflection.

The work presented in this paper can serve as a basis to motivate further exploration in a number of areas of research. From an educational perspective, further research should focus on the pedagogical considerations so that teachers can use, configure and maintain the overall multi-surface classroom ecology and the design and enactment of learning activities. Future research can also explore alternative and less intrusive methods to notify teachers about students’ problems in the classroom or to provide direct support to students. Usability studies may be conducted to identify the best ways to enhance teacher and students’ awareness and performance in the classroom. This may include consideration of vertical displays for each table so students and teachers can interact with multiple perspectives of the same content.

**ACKNOWLEDGEMENTS**

This work is partially funded by the Smart Services CRC.

**REFERENCES**


