Abstract
Brainstorming is a collaborative technique for enhancing creativity in problem solving. It has two main stages, storming and norming. To make the best of this valuable technique, learners need to build skills in the best practices for each stage. Tabletops have the potential both to support brainstorming and to help learners to build their skills in doing it effectively. This is important for those just learning brainstorming. It is also useful when any member of the group may have forgotten how to do it effectively. This paper introduces a scripting approach to support this learning. Our contributions are the identification of a three stage brainstorming process to supporting learning and the design of a script-based interface to aid the brainstorming process at each stage.

Introduction and Background
Brainstorming is a technique to enhance creativity and to help small groups articulate ideas. The central mechanism of brainstorming is that first, a large quantity of novel ideas is being created while judgment of those ideas is being delayed to a later point in time. This paper aims to improve support for the two key phases of this technique. In the initial storming phase, people should generate ideas, without critique, welcoming the unusual, combining and improving ideas and aim for quantity. Once this freewheeling stage is complete, the group moves to a norming phase, where it is important that all participants actively collaborate to assess and organise ideas [7].

Tabletops offer the promise of an excellent medium to support brainstorming. In early work supporting brainwriting [4], one person creates an idea and places it at the centre of the table for their partner to read and then take a turn at creating an idea. This conflicts with recommendations for free flow in the storming stage [7]. Recently Buisine et al. [1] compared two similar techniques, Mindmapping and one based on the original Osborn [7] brainstorming method.
Our Firestorm brainstorming system [2], the first tabletop system designed to faithfully follow Osborn, provided keyboard input to ensure quick idea generation, with ideas appearing in the middle of the table, colour coded to show the creator. An example tables shot at the end of the storming stage is shown at the top of Figure 1. Firestorm provided interaction elements to support the norming stage. The lower image in Figure 1 shows an example and Figure 2 shows people using these elements.

The design principles underlying Firestorm were based on a careful analysis of literature on how to make brainstorming effective [2]. These were:

- **DG1**: Support fast, concurrent idea generation.
- **DG2**: Many ideas visible on the tabletop at once.
- **DG3**: Don’t enforce orientations or territories.
- **DG4**: Foster awareness of ideas as they are generated.
- **DG5**: Make clear the creator of each idea.
- **DG6**: Support flexible grouping for idea convergence.
- **DG7**: Capture the group’s process and final outcome.

The evaluation study of Firestorm indicated that it was effective in terms of the number of ideas generated and user satisfaction. At the same time, some important limitations were identified.

1. It seems useful to introduce a new reflection stage, where participants consider what they have achieved, potentially reconsider the conceptualisation underlying the organisation, and consolidate their understanding of the group operation.
2. While the system followed **DG7**: Capture the group’s process and final outcome, it did not make use of this information to inform the group, or a facilitator, about the operation of the group.
3. Some individual participants and some whole groups failed to follow the instructions for the storming phase and this compromised the quality of outcomes.

Broadly, we concluded that tabletops have the potential to help people learn to use the brainstorming technique more effectively with more guidance and additional structure. To address Limitation 1, we propose the addition of a new stage as shown in Figure 3. To address Limitation 2, we propose to exploit the captured data in two ways. One is to capture a set of results so that groups can revisit parts of the process without loss of previous outcomes. Secondly, we will provide additional interface elements to support reflection. Both of these can include mechanisms to travel back in time to key stages. For example, the group may return to the end of the storming phase to consider a different conceptualisation of the ideas. Figure 4 shows the history of a group at set time intervals from the end of the storming phase through a top-down approach to norming, where they create categories and then place ideas into them. By contrast, a bottom-up approach would group ideas and only later identify labels for them.

An example of Limitation 3 was where one participant began playing with the tabletop during the early storming phase, disrupting the rest of the groups’ free flow of ideas. If the tabletop were disabled in this stage, with the only action being the generation of ideas at the keyboards, this disruptive behaviour would not be possible. Another example of Limitation 3 occurred where groups had previous experience of mindmapping and ignored the instructions to generate ideas quickly without any comment of judgement on them. To address this, we consider that it would be useful to help participants consider carefully each element of the recommended approach at each stage.
A promising approach to address Limitation 3 is to introduce collaboration scripts. These “promote productive interactions by designing the environment such that suggestions of different degrees of coercion are made” [5] which encourage specific activities that otherwise may not occur. Scripts provide an effective scaffold for learning and represent procedural knowledge and heuristics which form specific activities for which participants need to engage in. Scripts complement people’s internal scripts, such as a checklist of things to do, with explicit actions for moving along in a process.

**Scripted Design**

Our design approach aims to gain the potential benefits of scripts to “(1) Regulate learning activities, (2) Provide complementary procedural knowledge, (3) Provide process-oriented instruction, (4) Alleviate coordination, and (5) Foster awareness” [9]. At the same time, we aim to avoid the acknowledged pitfalls of scripts. Notably, overscripting with overly coercive scripts may dampen student motivation [8]. Interference with learners’ own effective scripts can cause problems [6]. Script adaptation may be needed as learner’s knowledge advances [9]. Script mismatch between the system and the learner context may lead to momentary lack of support (under-scripting) or interfere with the developing scripts of learners [3].

The key design principle to avoid these limitations is to provide user control of the scripting. Essentially, while we aim to identify the interface restrictions and advice that are appropriate for each of the three stages of the brainstorming process, we give the participants ultimate control over whether to enable or disable each of them. We anticipate that the decisions associated with this will enable the group members to discuss each aspect. This could help individuals learn about the recommended process for each stage. Support for this could be provided in tutorial information available at the table. It could also come from discussions within the group.

We now describe the design of the interfaces for controlling the scripting options for each of the stages shown in Figure 3 in terms of the prototype designs shown in Figure 5.

- **Storming:**
  - Disable table. This follows the recommendation that the group should focus on generating ideas in a free flowing manner during storming. Only the physical keyboards operate until the end of the phase. In the figure, the default toggle value is set, with the option preventing use of the table.
  - Enable colour coded ideas. This makes the system colour code each idea to show who created it (as in the tables for Firestorm). This increases accountability for the extent and nature of contributions. We used this in Firestorm, following DG5 to reduce loafing. However, one may argue that this may make people feel inhibited, so reducing creativity.
  - Enable facilitator. This allows the tabletop to provide advice according to set rules. For example, when the volume of ideas begins to decrease, the facilitator can issue suggestions to help generate ideas. In the figure, it has been disabled.
  - Advise to move to the norming phase. This allows the system to suggest a move to the next stage of the brainstorming process. In the figure, it has been disabled.

- **Norming:**
  - Top Down Approach. Groups first identify categories and then move ideas into them. The tabletop provides visual warnings for groups deviating from the method by just grouping ideas. This is the default.
– Bottom Up Approach. Groups group ideas and later identify labels for each group.
– Make new ideas appear near the creator. During storming, ideas appear at the centre of the table, but this option for norming makes new ideas and category labels appear near the creator, ready for discussion and placement.

Reflection:
– Self-assess. Activates interface for each participant to rate the quality of ideas and their organisation. This is a prelude to deciding to take more time to work through the following approaches.
– Restart Norming Phase in the opposite approach (Top down versus Bottom Up). As the group in Figure 5 had done Top Down Norming initially, the reflection phase in Figure 5 shows this option as regroup with Bottom Up.
– Try regrouping ideas for a different organisation. Allow a group to go back and redo the norming phase, perhaps after a period of discussion.
– Per User Analysis. The system provides information of the level of each individual’s participation.
– Group Analysis. Corresponding information for group.
– Compare approaches. Allows return to this and other group’s previous brainstorms on the same topic.

Key goals of this scripting is to help participants follow recommended techniques and better understanding group processes.

Conclusions
This work aims to improve support for learning brainstorming. It blends CSCL research on scripting with user control and systematic design of a brainstorming system. We have identified a new reflection phase and designed prototypes for dynamic scripted collaboration. These offer promise of increased awareness of how to brainstorm and collaborate.

Acknowledgements
This work is partially funded by the Smart Services CRC.

References