Heuristics to Support Design of New Software for Interaction at Tabletops

Trent Apted
School of IT
University of Sydney
tapped@it.usyd.edu.au

Anthony Collins
School of IT
University of Sydney
anthony@it.usyd.edu.au

Judy Kay
School of IT
University of Sydney
judy@it.usyd.edu.au

ABSTRACT
Heuristic Evaluation is a “discount” usability test that can support improved design decisions early in the development cycle. It is particularly a technique to support software design for tabletop interaction because the field is so new and there is potential to explore many new ideas for interaction. This makes Heuristic Evaluation particularly valuable. Many sets of heuristics have been proposed in the past, both for interfaces in general (not just computing interfaces), and ones that are more specialised. For tabletop, and other horizontal interactive interfaces, current sets of heuristics fall short. In this paper, we build from previous sets of relevant heuristics, to formulate a new set of heuristics for software design for tabletop interaction.

Author Keywords
Tabletop interface, single display groupware, collocated collaboration, heuristic evaluation

ACM Classification Keywords
H.5.2 Information Interfaces and Presentation: User Interfaces—Evaluation/methodology; H.5.3 Information Interfaces and Presentation: Group and Organization Interfaces

INTRODUCTION
There are fundamental differences between tabletop interfaces and conventional desktop computer interfaces. The tabletop provides both opportunities and challenges of innovation in creating new ways to interact. If we are to exploit the opportunities offered by the special properties of tabletops, whilst taking account of the limitations that they impose on design, we need support for the exploration of the rich design space. This must also take account of the current state of flux for tabletop hardware, with many different sized tables, limitations of single-touch in some, the possibility of distinguishing users in some, and different input possibilities including finger(s) and stylus.

To facilitate this design process, we need techniques to assist in evaluating a range of possible ways to design new software elements for tabletop interaction. With the current early state of the evolution of tabletop interaction, there are many interface elements that deserve exploration. Existing systems have many different ways to do even such primitive actions as moving, rotating and resizing objects visible on the table. Many more interaction primitives are yet to be explored. For example, there has been little work on a mechanism for accessing files that are stored in a file system attached to the table. Nor has there been exploration of such basic actions as copying elements on the table.

Heuristic Evaluation was introduced by Molich and Nielsen [6] to ameliorate the requirement for expensive usability engineering. Human-computer dialogue construction appears deceptively simple, yet it is full of subtle pitfalls (ibid.). The design issues are also more significant when supporting human-to-human interaction through collaborative interaction. Heuristic Evaluation has the very desirable property that it can be conducted quickly, and relatively inexpensively: suitably expert evaluators can make use of a set of heuristic guidelines, or rules, to evaluate an interface. Consequently, future software for tabletop interaction will be enhanced if we can use Heuristic Evaluation as part of the iterative design process. Although there are many sets of guidelines, some extensive, such as MITRE [19], and others highly specialised, there are none that properly support the particular class of tasks involved in designing new interaction elements for tabletop interaction. Certainly, the very broadest design guidelines, which apply to all interfaces are still of value for tabletops. However, established heuristics fall short of some critical interface challenges for tabletops. It is these shortcomings that this paper aims to establish.

To begin, and to put the heuristics in context, we identify the distinctive aspects of tabletop interaction. This is followed by an overview of existing heuristics and how they fall short. From this we identify a set of heuristics for evaluating software designs for tabletops.

DISTINCTIVE ASPECTS OF DESIGN FOR TABLETOPS

The tabletop, while offering many benefits for collaboration interaction, imposes new challenges over those already present in conventional interface design. Here we outline the distinctive elements of tabletop interaction that must be taken into account for effective design.

Collaborative Interaction: A key goal of tabletops is to support collocated collaborative interaction. As tables fit the normal way that small groups of people often choose to socialise or work together, tabletop interfaces should support multiple people working at the table, possibly with multiple goals. However, interfaces should also support individuals making use of the tabletop. It is particularly important to design for small groups of people: this is partly because of table and room size, as well as typical limitations on hardware for the near future, but also because many activities that
require collaboration involve small groups.

**Context of Use:** The usefulness of tabletops for collaboration suggests that they will often be located in shared spaces, rather than private. This means that there are more serious barriers to the personalisation or customisation that is common for desktop computers. It may also make it particularly important to design for the novice, irregular or intermittent user, making learnability particularly important.

**Orientation:** A fundamental facility of a collaborative tabletop interface is to support people sitting face-to-face and around-the-table, and so tabletop interfaces must support arbitrary orientation of interface elements [17]. Interface elements must also be orientation independent when presenting information, as a key concern is state ambiguity induced by viewing the table from different angles [13].

**Tabletop Size:** This has an impact on a number of design factors, such as the size at which to display text. In addition, tables vary in size considerably more than traditional computing interfaces, which usually differ only by a few inches. Consequently, interface design should easily adapt to varied display size. At the same time, there is a normal upper bound on the size of typical tables.

**Human Reach:** As tables vary in size, the limitations of human reach and direct-touch interaction must be considered, as a user may not be able to reach some tabletop elements. Designers must also consider social expectations — a user may potentially interfere if they must reach into an area when working at the tabletop [14].

**Use of Table Area:** Display of system messages, such as confirmation dialogs, may need to be unobtrusive so as not to interfere with other users. However, they should not be totally in the periphery so as not to be noticed (particularly if the table is large), or else noticed only by the incorrect user. Users will also tend to establish personal, group and storage areas when working at the tabletop [14].

**Clutter:** When considering table size, display resolution, and multiple users, clutter is a critical problem to address. Clutter may arise from the need to present large amounts of information [16], or the need to replicate information for multiple people [8]. Shared and individual objects on the tabletop may overlap, particularly when they are zoomed-in for a more detailed view [12]. In a multi-user setting, management of clutter becomes challenging — existing methods in conventional personal computers, such as a ‘task-bar’ to control active windows, were only designed for single-user interaction at a fixed position on the screen.

**Limited Input:** It must be natural to interact with the tabletop using input with special constraints not present in desktop computer interaction. A keyboard and mouse is typically not present in a multi-user tabletop setting, and so interface selection targets must be sufficiently large for direct-touch interaction, which potentially contributes to interface clutter. While providing a virtual keyboard on the tabletop is possible, this provides no tactile feedback when pressing keys, and does not give users a fixed reference of where to place their hands. The interface must also be resilient to accidental touches, and gesture ambiguity (e.g. whether a touch with two fingers together is interpreted as a large finger or multiple fingers). One must also consider user habits from regular tabletops, such as leaning upon them, placement of documents and utensils, and clutter of physical items.

**EXISTING HEURISTICS AND GUIDELINES**

Given the distinctive elements of collocated tabletop interaction, we now outline and discuss existing heuristics from domains that share common characteristics with tabletop design.

**General Heuristics**

There are several established general heuristics, for example Shneiderman’s [18] “eight golden rules of dialog design” or Nielsen’s [9] ten usability principles for interface design. These rules and principles are important for an effective human-computer dialogue. Nielsen’s heuristics are: 1) Visibility of system status; 2) match between system and the real world; 3) user control and freedom; 4) consistency and standards; 5) error prevention; 6) recognition rather than recall; 7) flexibility and efficiency of use; 8) aesthetic and minimalist design; 9) help users recognise, diagnose, and recover from errors; and 10) help and documentation.

These heuristics are clearly applicable to tabletop interface design. Notably, they can take on new meaning in a tabletop setting. For example, (2) is often phrased as “speak the user’s language”, but for tabletops we can extend the match to the users’ real world by leveraging direct interaction with physical metaphors. However, while the rules are generally applicable, they do not address the distinctive elements of tabletop interaction.

**Groupware Heuristics**

It is natural to build from work on CSCW since that has one key common element — the goal of supporting group interaction at the interface. Despite the major focus on distributed collaboration in CSCW, it does provide some useful work on heuristics for design for collocated collaboration. Baker et al. [2] argue that a problematic aspect of existing heuristics is that they focus on taskwork aspects of an interface, rather than the teamwork aspects — a concern that would be shared with tabletop user interface design. Baker et al.’s groupware heuristics are: 1) Provide the means for intentional and appropriate verbal communication; 2) Provide the means for intentional and appropriate gestural communication; 3) Provide consequential communication of an individual’s embodiment; 4) Provide consequential communication of shared artifacts (i.e. artifact feedback); 5) Provide protection; 6) Manage the transitions between tightly and loosely-coupled collaboration; 7) Support people with the coordination of their actions; and 8) Facilitate finding collaborators and establishing contact.

Some of these heuristics are relevant to collocated tabletop interaction. Item (7) is important for tabletop interaction, particularly for hardware that is not multi-touch. Despite some similarities, groupware heuristics alone do not address many of the physical factors of tabletop systems.

**Large Display Heuristics**

We considered large displays because tabletops are often similar to these in terms of the display technology and the limitations in interaction, especially input. At the same time,
there are important differences, notably that large displays on walls do not have the orientation issues of tabletops. A large display generally does not share the same direct-touch interaction properties as tabletops, meaning that the limitations of human arm reach are not a restricting factor.

Somervell et al. [20] identify a set of heuristics for Large Screen Information Exhibits (LSIEs). They argue that other researchers [2 5] have not adequately described their approach to developing heuristics and so use Scenario Based Design (SBD) [10] and a review of five existing LSIEs, to determine a set of potential heuristics. The heuristics proposed by Somervell et al. are: 1) Appropriate color schemes can be used for supporting information understanding; 2) Layout should reflect the information according to its intended use; 3) Judicious use of animation is necessary for effective design; 4) Use text banners only when necessary; 5) Show the presence of information, but not the details; 6) Using cyclic displays can be useful, but care must be taken in implementation; 7) Avoid the use of audio; and 8) Eliminate or hide configurability controls.

These include cases that conflict with tabletop interaction design. Item (2) states topic information should use categorical, hierarchical or grid layouts. In tabletop interfaces, screen space needs to take into account orientations of users and personal spaces, making the presentation of fixed structures problematic. Although (5) is relevant to tabletops, Somervell et al. state that viewing of detail is better suited to desktop interfaces, and LSIEs should only provide an overview to information. The large workspace of a tabletop, potentially offering both shared and private workspaces to users, affords greater possibilities to interact with more detailed views of information, but interface clutter must be considered. It is also unclear whether (7) is valid for tabletop interaction.

Czerwinski et al. [4] identify six broad categories of usability issues in a research overview for large displays. Here, their emphasis is on large, or multiple, desktop displays running Microsoft Windows, rather than wall-sized displays. The usability issues identified are: 1) Losing track of the cursor; 2) Distal access to information (e.g. accessing Start menu); 3) Window management problems (e.g. placement across bezels); 4) Task management problems (e.g. increasing numbers of windows); 5) Configuration problems (e.g. when there are multiple monitors); and 6) Failure to leverage the periphery (e.g. supporting peripheral activity awareness).

While (1) often does not apply with direct-interaction tabletops, we have seen cursors used to support remote collaborators on tabletop interfaces (e.g. [3]). Item (4) is suggestive of problems of clutter commonly found on tabletop interfaces.

Tabletop Heuristics
Scott et al. [15] conducted an analysis of existing tabletop research from the HCI and CSCW fields. From this, they propose a set of heuristics to facilitate the design of tabletop interfaces, and outline directions for future research. The heuristics proposed are: 1) Support interpersonal interaction; 2) support fluid transitions between activities; 3) support transitions between personal and group work; 4) support transitions between tabletop collaboration and external work; 5) support the use of physical objects; 6) provide shared access to physical and digital objects; 7) consider the appropriate arrangements of users; and 8) support simultaneous user actions.

A clear focus of these heuristics is supporting transitions (such as between activities or people), rather than usability. Moreover, they conflate issues related to hardware and those under control of the software. As such, the purpose of these heuristics seems to lie with the design of the overall user experience, at a high-level. They cannot inform design of specific software interaction elements, making them inadequate for the important design yet to be done exploring different interface widgets and gestures.

TABLETOP SOFTWARE HEURISTICS
We have outlined the distinguishing characteristics of tabletops, and reviewed key sets of heuristics. While these have some applicability to tabletop interaction design, they do not address the important, distinguishing properties of collaborative tabletop interaction. Based on our discussion of these properties, we propose the following heuristic for evaluating tabletop software interaction, along with their justification.

Design independently of table size Design for different tabletop sizes and allow flexible resizing of all interface elements.

Interfaces should not be constrained to a particular table size, and designers must consider that some interface elements may need to be regularly enlarged depending on the task or to be legible by all users (particularly those with restricted eyesight or the elderly [1]).

Support reorientation* Allow all interface elements to be easily rotated to support users working at any position around the table, and consider users moving around the table while using it.

People should be able to view and interact with the table at any position around it, and interface elements should be easy to rotate (such as in [17]). Adapted from (7) of Scott et al.’s [15] guidelines.

Minimise human reach Consider that users may not be able to physically reach all interface elements. Elements must be moveable to all areas of the table.

Interface elements should not be fixed, as that could constrain users’ positions at the table, and cause usability problems if some elements are unreachable. Designers must also consider social expectations that prevent users from reaching in front of others [17].

Use large selection points Design independently of tabletop input hardware, but support large input cursors (e.g. human fingers) where possible.

To provide a natural interface with the restricted input available at tabletops, interface elements should be usable through direct-touch interaction with large fingers or other input stylus.

2Heuristics adapted from Scott et al.’s [15] tabletop guidelines have been italicised and marked with an asterisk.
Manage interface clutter  Support quick removal or hiding of objects on the tabletop, while ensuring management of clutter by one user does not have unwanted side-effects on other users of the table.

Clutter management is a significant problem to address in tabletop interface design [14], due to constraints on display and input technology, table size, and supporting multiple people working concurrently.

Use table space efficiently  Avoid modal behaviour that limits the utilisation of table space. Allow arbitrary groupings of interface elements for personal and group spaces.

Modal behaviours, such as confirmation dialogs that take focus away from the rest of the table, constrain multi-user collaborative interaction. It is also important to support people forming personal and group spaces—a natural tendency when collaborating at a tabletop [14].

Support private and group interaction*  Support interaction by a single user or multiple users.

Interface elements should be usable by a single user, or used as a shared resource by multiple users, possibly with different goals. Adapted from (3) and (8) of Scott et al.'s [15] guidelines.

These heuristics provide a foundation for evaluating tabletop software interaction, though not all of them are applicable to some tabletop designs. For example, if a particular tabletop is designed such that people may only stand side-by-side, supporting reorientation may not be important. Furthermore, some existing general heuristics, such as Nielsen’s [9], are also applicable to many aspects of good tabletop interaction design (such as error prevention). The proposed heuristics have value in augmenting Heuristic Evaluation with existing general guidelines, to ensure that the distinctive properties of tabletop interaction are more thoroughly addressed.

CONCLUSION

Adoption of tabletop computing is imminent. As yet, there are no clear guidelines to help us design an interface that retains usability on a tabletop. Instead, tabletop interfaces so far have relied mainly on researcher intuition for the design, and much of this intuition and findings from preliminary user trials goes unpublished. There is also no established paradigm for tabletop interaction (e.g. WIMP — Window, Icon, Menu, Pointer) from which to draw a common ground. Future software for tabletop interaction will be enhanced if we can use Heuristic Evaluation as an integral part of the iterative software design process.

We have presented an overview of distinctive elements to interaction at a tabletop. These elements reveal the inadequacy of applying general heuristics to tabletop interfaces. We have reviewed heuristics for interfaces which have some aspects in common with the tabletop, though these fall short in addressing all important properties of collaborative tabletop interface design. To address these limitations, we have proposed a set of heuristics that can be used to ensure that future interface designs address the constraints of tabletop interaction, and take advantage of the new collaborative possibilities. We hope that this work will initiate a discussion of the value of Heuristic Evaluation for tabletop innovation.

ACKNOWLEDGEMENTS

This work is partially funded by the Smart Services CRC.

REFERENCES