Pushing the limits of ATC user interface design beyond S&M interaction: the DigiStrips experience

Christophe Mertz (+,°), Stéphane Chatty (+), Jean-Luc Vinot (*)

(*) CENA
1, rue Champagne
91200 Athis-Mons, France

(+) CENA
7 avenue Edouard Belin
31055 Toulouse, France

(°) CR2A-DI
32, rue des cosmonautes
ZI. du Palays BP 4067
31028 Toulouse cedex 4

{mertz,chatty,vinot}@cena.dgac.fr

ABSTRACT

Most designs proposed for air traffic control workstations are based on user interface designs from the early 1980s, though research in user interaction has produced innovations since then. Project Toccata federates a series of research held at CENA on new user interfaces and services for ATC. Virtuosi and DigiStrips are two prototypes developed for Toccata, which make use of touch screens and served as a basis for research on the use of graphical design techniques in user interfaces. This paper describes the lessons learnt in that experience and argues that techniques such as animation, font design, careful use of graphical design techniques can augment the possibilities of user interface design and improve the usability of systems. We finally analyze the possible implications on ATC workstation design.

KEY WORDS: touch-screen, animation, graphical design, feedback, gesture recognition, informal assessment, mutual awareness, electronic flight strips.

INTRODUCTION

In the last decade, many western countries have embarked into projects aimed at modernizing their Air Traffic Control (ATC) workstations. It appears that most of these projects have sticked to early design choices made on knowledge and technologies that are now quite dated, though still little validated through operational experience. Fifteen years ago the computing industry was discovering Wimp interfaces (Windows, Icons, Menus, Pointing). The ATC research community soon enjoyed dealing with Wimps, and progressively developed a common culture based on S&M interaction (Sony screens and Mouse). Most S&M systems share the same features: plain and strict appearance, coarse interaction styles (with the occasional use of rubber band techniques), indirect interaction through rigid devices (mostly a mouse), absence of visible feedback, heavy use of menus and windows. In Europe, examples of hard-core S&M systems include early prototype versions of ODS-France and the ODID prototypes [see graham97].

Meanwhile, the more general computer-human interaction community has gained considerable knowledge, and has identified the weaknesses of Wimps. A number of experimental results highlight how S&M interactions can be painful compared to softer modern interaction styles. At the same time, computer-human interaction has definitely evolved into a design discipline. For the last five years, we have pursued that approach, striving to design elements of future ATC workstations that could be made of touch-screens, flexible interaction styles, and elaborated graphics designs that convey subtler feelings than S&M interaction.

The Toccata project currently under way at CENA federates the results of several research projects carried out during those years. Some of those projects have involved formal performance evaluations of interaction styles, whereas others have more focused on design and innovation. In this article, we will concentrate on the latter, with a special focus on two issues: the importance of graphical design applied to all displays and feedback, and the design of touch-screen-based interfaces. We will use our prototypes Virtuosi and DigiStrips to illustrate our discussion of those issues, and the guidelines we propose for designing touch-screen interfaces. We will argue that careful and well-informed design makes a difference, and that it can dramatically improve the usability and the acceptance level of a product.

RELATED WORK

In many ATC centres, air traffic controllers currently use a system that is little if at all interactive, apart from the settings of their radar screens. The essential work aid is a set of paper strips, called flight progress strips. Strips are printed by the computer system, controllers manipulate them to organize their work and annotate them to support their memory. In the last decade, there have been two main approaches to modernizing that system. Both approaches take advantage of the necessary modernization of radar screens, which are being replaced by very large color computer displays. They are based on interaction technologies that have been popular since the early 80's: mouse, pointing and windows. This combination of a large screen (usually made by Sony) and mouse interaction is what we call S&M interaction.

The first approach consists in trying to transpose paper strips into the computer system, leading to what is commonly called 'electronic strips'. These usually are faithful but stylized computer representations of paper strips and their contents, and they are manipulated with the computer mouse. Such electronic strips are either displayed on the large radar screen or on a horizontal secondary display. At least three types of manipulations are possible on paper strips: handling them, writing on them, and referring to them during conversations or gestures between controllers. Writing is generally simulated by menus. [mertz97] compares different menu styles applied to ATC and compares their performance to keyboard input with subjects under controlled cognitive load. Handling poses even more problems: in France for instance, moving strips around with a mouse has been rejected at early evaluation stages [garron92]. This led engineers to try and imagine automated strip layout algorithms, with little success [labarthe94]. And finally, gestural, non-verbal communication was either forgotten or recreated through painful mouse dialogs and color-coding.

The second approach consists in doing away with paper strips, by providing direct interaction with aircraft representations on the radar screen and displaying substitute representations. It is the approach taken by Erato [leroux98] and the PHARE Demonstration 3 [chabrol99]. Erato has been quite successful in providing a usable and efficient substitute environment, though a few interaction weaknesses remain [marais99].

Beyond their opposition in terms of services proposed to controllers, the common point between those two approaches is the use of the most classical WIMP technology. Before mentioning radically different approaches, we will survey the recent evolutions of the Human-Computer Interaction discipline related to that topic.

Evolutions in HCI

Human-computer interaction has been a fast-evolving research field in the last 15 years. As regards graphical interfaces, the precepts of direct manipulation introduced in 1983 have been refined and are now more widely understood: misconceptions such as considering menus a good example of direct manipulation are now rare, for instance. Beside those qualitative notions, a set of more quantitative models and theories has emerged. The notion of usability has been defined and refined. Models of interaction such as GOMS serve as a framework for evaluating the performance of user interfaces. Predictive models such as Fitts' law or the steering law [accot] provide a better understanding of the performance and limits of interaction styles. Based on that knowledge, a number of increasingly efficient interaction techniques (or styles) have appeared for different types of media. Marking menus [kurtenbach95], Hotbox [kurtenbach99], and rhythmic menus [maury99] are examples of efficient interaction styles usable with a mouse. But as the limits of mouse interfaces and more generally of direct manipulation have appeared [buxton], other technologies have been explored: pen computing and multimodal interaction are increasingly well understood, for instance. Direct interaction on touch screens is increasingly popular [shneiderman93, p.187], especially with the appearance of large (18" to 20") flat touch screens of good display quality.

At the same time, the evolution of technology has allowed for entirely different schools of thoughts to emerge. On the one hand, virtual reality provides many ways of immersing users into 3D simulated worlds. But applications are thus far limited to entertainment or very extreme situations. On the other hand, tenants of augmented reality suggest that technology become more ubiquitous and immersed into real world objects [wellner93]. This has applications at home, school, and in work environments. Related research explores the use of "spatial multiplexing", i.e. the use of multiple ad-hoc input devices of a given task [kurtenbach97, fitzmaurice97, rekimoto98].

Alternative approaches in ATC

A few recent projects in ATC research have common points with the evolutions in HCI. An apparent paradox of those projects is that they often have close relationships with older ATC workstations. This is mainly because, like the most recent research in HCI, they try to take advantage of users' ability to spatially multiplex their actions when several physical interactors are available instead of a sole mouse. The STARS project for instance was partially redesigned to move a number of display controls from the WIMP interface to real physical knobs and buttons [faa98]. Similarly, Grigri [chatty96] explored how pen computing could be applied to ATC in order to restore the pen-based interaction French controllers are used to. The Cameleon project [mackay98] explored possible applications of augmented reality techniques to paper strips, ending up with hybrid (half-paper, half-electronic) strips.

In the same way the research we describe in this article is similar in some ways to older systems: touch screens are an important part of the Eurocontrol Maastricht ACC user interface, as well as of the French flight plan management system.

THE TOCCATA PROJECT

Over the last five years, CENA has carried out research on the performance of user interaction techniques and their potential applications to ATC: prototypes of two-handed input [chatty94, puleggi98] or pen-based computing [chatty96], performance evaluations of menus [mertz97], and even theoretical contributions to interaction models [accot99]. It was recently decided that the results of those researches needed to be federated in a workstation demonstrator: Toccata. The goal of Toccata is to demonstrate or evaluate a whole series of interaction styles, display techniques and communications tools. The prototyping environment Fugue provides the flexibility required to run those demonstrations, which is now publicly available as free software [chatty00].

Though Toccata aims at testing many variants of user interfaces, the demonstrator needs some form of architectural consistency to be credible. The architecture we chose is like the recent research in user interfaces for ATC we described earlier: closer to earlier designs than to recent S&M interfaces. Our reason for that is experimental. Considering the intrinsic limitations of the mouse in terms of performance and flexibility, and the risk of overcrowding the radar display with too much information, we want to explore the possibilities of a more traditional architecture: use the vertical area of the workstation for information display and the horizontal area for work aids. This implies the use of touch screens for the horizontal area, and we are currently testing different configurations of touch screen size and quantity: from one to three or more screens, and from 14” to 20”.

This architectural choice raises a number of research questions: what kinds of tools to provide on the touch screens? What interaction techniques for supporting note taking and data input? How to relate the aircraft representations on the radar display and on the horizontal surface? The two prototypes that we use in this article have been designed to explore some of these questions. Virtuosi proposes an alternative to menus for taking notes and entering data into the system: first contact, clearances, and even unexpected requests or events about the aircraft. DigiStrips is a revisiting of the old problem of electronic stripping, used to support research on the possibilities of touch screens and the impact of graphical design on the usability of user interfaces.

We used the development of Virtuosi and DigiStrips to explore questions about the design of user interfaces. On the one hand, there is little literature on designing for touch-screens, and we needed to build some knowledge on that topic. On the other hand, we had a strong feeling that graphical design could bring a lot to user interface design but once again there is little literature about that. In this article, we report about the lessons learnt and the design guidelines we derived from that experience, starting with graphical design.

GRAPHICAL DESIGN AND ANIMATIONS

Appropriate fonts and good graphical design can increase displayable information:

By using dedicated and carefully designed fonts and by precisely composing the graphic components, it is possible to display more information while making it more legible. For instance, one of the findings of DigiStrips is that is possible to display almost all the printed information of French paper strips for up to 30 strips on a 20" screen.

Texture or color gradation can code information:

The work described in [metcalfe98] was innovative and proposed to use transparency for background information or airways on radar display. However for the rest of the interface they only used infills, lines and texts and they did not consider using textures or color gradation. One reason may be the limited number of colours (usually 256) available on most 28” screens, though this is now outperformed even on home computers. [graham97] states "that colors may reduce the ability to build a traffic picture". This seems very strong a statement and in facts it is probably influenced by the use of large infills of saturated red as conflict coding. Similarly, [cardosi99] claims that "the number of colors assigned a different meaning should be limited to six". But such a statement is often interpreted in an abusive way, limiting the total number of colours available on the screen. Colours can be used in many other ways than for state coding, though. For example, in Virtuosi (figure 1), current selected instructions appear clearly, on the foreground, thanks to color gradations. This color-coding is relatively self-explanatory and should require little training and little cognitive effort to remember. Similarly, in DigiStrips a different texture codes the zone where the user can handle the strip and move it (approximately the left third of the strip[1], see figure 2). This subtle coding is enough to remind or show the user the limits of the grip zone.

Figure 1: current selected instructions displayed in Virtuosi

[1] The difference of textures may be difficult to see on black&white low quality prints of this paper.

Different fonts can convey information:

ATC system designers are often reluctant to use fonts to code information. For example it may be a bad idea to code the "assume control" state of a flight with two fonts or with their slant or bold attributes. This type of coding may indeed be difficult to memorize and even to recognize. However, it is possible to distinguish system-computed data and user input data through the font. For the former, we used "computer fonts" and for the latter we use legible "hand-written fonts". Such a coding appears easy to perceive, understand, and remember. In figure 2 for instance, it is easy to detect with a glance which flights have been given clearances.

Figure 2: strips with hand written font for 'annotations'

Similarly, [marais99] mentioned that the Transfer Flight Level (TFL) should be distinguished when it is a standard pre-computed value or when the Planning Controller modified it after a negotiation with the following sector. Coding it with a "script-like" font seems a good choice. Other levels of coding can probably be imagined, as fonts can convey a number of subtle yet well-perceived nuances. For instance, Downlinked Aircraft Parameters could be distinguished from radar data on a radar display by using a "digital-watch" font.

Animations

Like graphical design helps displaying more information, animations in interfaces are useful in expliciting state changes. They can be used to explicit the result of an action as on the Macintosh desktop when the user sends a document into the trashcan. They can reveal events like the arrival of an email in your mailbox. They can even complete an action or give an indication of job advancement. Animations can also be used for alarms [athenes00] (we will not discuss this use of animation in this paper). Animations can be very useful in an ATC interface, and we will give some animation examples from DigiStrips and Virtuosi prototypes.

Animation can facilitate transitions:

The user can move, push or shift DigiStrips flight strips. When a strip is moved between two others, they move altogether to clearly display the result through animation. User thus clearly perceive the result of the action instead of reading, memorizing, reading again, and comparing call signs of electronic strips, which is what they need to do without animations. Such animations allow users to quickly detect what is wrong in case of minor interaction errors. The observations we did during the performance evaluation (described later in this paper) demonstrated it. Similar animations can also be used for managing messages or flight lists.

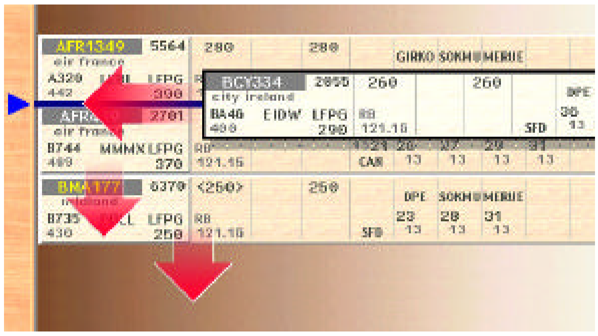
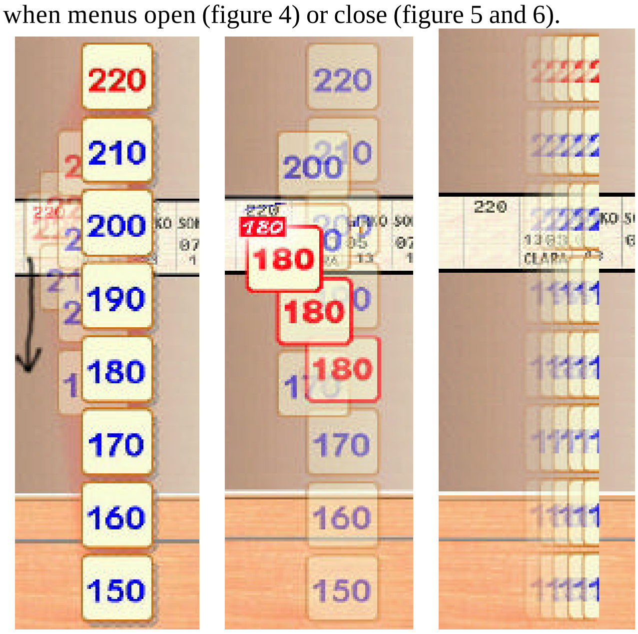


Figure 3: animation following the insertion of a strip

Animation improved menu opening or closing:

[marais99] reports that users miss feedback at the end of input with a menu. Users felt sometimes unsecure about which flight in the radar display the menu applied to. In DigiStrips we tried some short and simple animations when menus open (figure 4) or close (figure 5 and 6).



Figures 4, 5, and 6: menu opening, menu closing and menu canceling in DigiStrips

This has the following advantages:

?? the menu-opening animation re-inforces the feedback on which flight the input will apply: opening begins in the strip, and the menu grows from this position;

?? a menu can hardly open unnoticed (due to an involuntary "click" or gesture on the touch screen). The user has a chance of seeing it with his or her peripheral vision;

?? at closing, the menu shrinks toward the strip on which the input applies. This feedback helps the user perceiving he or she did not mistakenly input data on a wrong flight; while the menu shrinks, the colored selected value moves (during the animation) towards the modified field. Finally, the new input value is animated during approximately 0.5s. All these animations allow the user to easily see and perceive both the input value, the modified field and the modified flight. These animated “system-acknowledge” are similar to the pilot acknowledgement. It greatly helps the user verifying his or her input. The user becomes more system-confident.

if the user cancels the input (by touching the screen outside the menu) the menu "closes" from the left to the right, to clearly state that no input has been made. This avoids ambiguity about whether there has been a mistake or not.

These kind of animated menus could also be used in S&M interfaces. For instance, menus on radar display could be animated just alike.

Animations can enhance scrolling menu:

In most S&M interfaces, menus used to enter data are scrollable. For example the Cleared Flight Level opens on current flight level (or expected Exit Flight Level?) and the controller can scroll up or down if the value he wants to input is not displayed. Scrolling is instantaneous. We used animations in a similar case in Virtuosi, where the horizontal bars are like big, horizontal, scrollable menus. The left and right arrows let the user scroll around if necessary, and the values move (right-bound or left-bound). They accelerate and then decelerate to show a new set of values. The user can even interrupt the scrolling animation if he sees the desired value moving. We also use a bouncing animation to notify the user when the scrolling bar reaches its end. DigiStrips offer simple linear animation when the user scrolls through a Flight Level menu by re-drawing vertical signs on the flight strip.

Animations can notify events:

Currently with paper strips, the controller is noticed by the slight noise of the printer. In S&M, new strips, new messages or new elements in flight lists are barely notified to the user. In DigiStrips, we animate the new strip when it arrives. It moves from right to left at the bottom of the touch screen. The user notices it seamless.

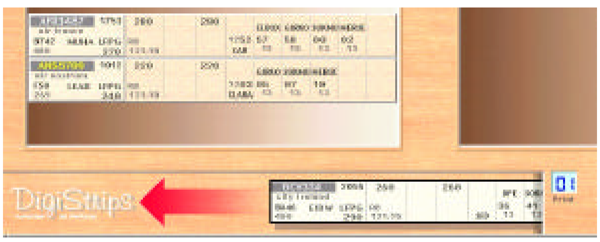


Figure 7: arrival of a new strip in DigiStrips

Animations can make explicit the effect of an unfinished action:

In DigiStrips we used animations to show what will happen if the user releases a strip over the trashcan. As soon as the strip is close enough from the trashcan, slight collapsing rectangles show that the strip will be trashed if released.

Animations can help when system reaction time is too long:

When complex tools are used as in the Phare Demonstration 3 [chabrol99], it is not always possible for the interface to give immediate feedback of success or failure of a request. It was the case for example for the Arrival Manager when the user moved a flight in the departure sequence. She had to wait 2 or 3 seconds before the system accepted or refused the request. The CENA Team in charge with the development of the interface added animation (lasting approximately 2-3 seconds) when the system answered. If the user was not looking at departure manager she had a chance of noticing this animation and the result in her peripheral vision.

Animation can enhance mutual awareness:

Controllers usually work by twos or threes (in case of a student controller). There are even sometimes four or more controllers working together (for instance during stormy weather). The system’s ability to support this collaborative work is often underestimated. Animations can help by making more explicit to the other ones what a controller is doing. Though we do not claim that DigiStrips has the properties as paper strips regarding mutual awareness, animation does make a difference with other designs.

DESIGNING FOR TOUCH SCREENS

Designing for S&M or touch screens is different in many aspects. First the size of the finger pulp does not allow the same precision than a mouse. Then you are limited to the equivalent of a single button. But to balance that, you gain the ability to use gestures and you avoid too small graphical buttons, costly even with the mouse [mertz97]. A good guideline when designing for touch screen is to have in mind the objectives of pointing with any hand and using the application even when standing up. We will now give some advantages and some rules usable when designing touch screen based interfaces.

Why using touch screen

Touch screens favor gestures:

Touch screens are direct interaction devices [baber98]: you point directly on the graphical object, on the screen, not via an indirect peripheral like a mouse or a trackball. This means that the user may be able to interact with less visual attention in a semi-blind mode. He has to look at the screen where the target is located, and then he can point his finger on this target without tremendous visual attention. This is just impossible with a mouse, which gives less proprioceptive clues about movement. The user must visually track the mouse pointer.

Touch screens favor mutual awareness:
Because you see what your colleague is doing with his hand on a touch screen you get many clues on his activity. A controller can also point on the screen to show something to his college. He does not necessary need a dedicated "show" function, which was sometimes implemented in S&M interfaces.

Touch screens can be shared between users:
If you worked together with a colleague in front of your desktop, you must have noticed how difficult it is to share a mouse. With touch screens this becomes possible. Current technology does not allow simultaneous interaction on the same touch screen, but it is possible to share it and interact alternatively.

Designing for touch screen interfaces

Design with paper and video prototypes:
Using paper and video prototypes is easier with touch screen. Reasons are simpler display and interaction than with a mouse; the hand or the finger of a subject is easy to video-record. We used such paper and video prototyping for the design of DigiStrips [mertz98] and Virtuosi. It gave us quickly clues such as:

- ?? Avoid using buttons for choosing the manipulation function (move, push, and shift), but use limited and simple gestures if all users should be able to use it with both hands
- ?? move the object and not its ghost, as in most WIMP interfaces
- ?? animations to set strips at their precise place (just after the user manipulate a strip) create a natural interaction
- ?? even very simple and raw animations help the user

Consider take-off strategy:
With touch screens, even graphical buttons should behave differently than the "classic" motif toolkit. Potter in [shneidermann p.161] proposes a new touch strategy called take-off, which is more precise (but may be slower than others). The take-off strategy allows the user to drag her finger and the button is selected when she releases her finger from the touch screen (in motif toolkit the button selected is the one touched at first contact, but if the finger get out no button is selected). We used this take-off strategy for the selection of a value in Virtuosi value bars and in DigiStrips menus.

Allow the user to correct interaction micro-errors
When we designed the archive function by placing a strip in a trashcan (figure 8), we added a very simple undo function.

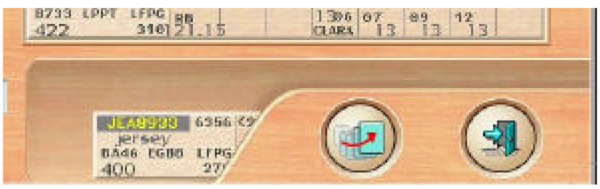


Figure 8: the DigiStrips trash can

The last trashed strip appears in the trashcan. If the user trashed the wrong strip, he does not have to open the trashcan and select the strip. He just withdraws the strip from the trashcan with his finger and deposits it in his working area.

Use very simple gesture recognition:
In the past we studied pen computing and gesture recognition applied to ATC [chatty95]. But designing letter-like or digit-like stroke, easy to draw by every user, easy to remember and easy to recognize is difficult if not impossible with current technology. However it is possible to implement simple and efficient stroke recognizer if these strokes are simple enough like those used in marking-menus [kurtenbach95]. These strokes should be drawable with a stylus or with the finger, with both hands, and even when the user is standing. Such simplified strokes (figure 9) can dramatically improve touch screen based interface.

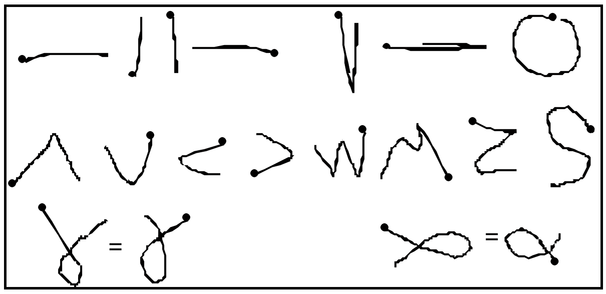


figure 9: simplified strokes recognized in DigiStrips (only some of them are currently used)

Different gestures can open different menus:
We just recently added input data in DigiStrips. We did it by combining simple gesture recognition (similar to markup menus [kurtenbach95]) to open different menus. For example, to open a Cleared Flight Level menu, we use vertical gestures drawn on the flight level strip. With gestures it is even easy to selected lower level values (downward straight stroke) and higher level values (upward straight stroke). To input headings, we use horizontal strokes. A left-bound stroke opens a turn-to-left menu and a right-bound stroke opens a turn-to-right-menu (figure 10). This dialog is quick and easy to remember.

Figure 10: the right-bound stroke opens a turn-to-right-menu in DigiStrips

Touch screen permits writing:
Many different touch screen technologies allows the use of a stylus (either passive or active depending on the technology). This allows an easy implementation of free writing input, without any recognition. The controller can write any special information and associate it to a flight, either to support his memory, or to benefit to the take

over or to be send through the system to the following sector. We currently demonstrate this function in Virtuosi (see top right of figure 1). We recently implemented an free-text input in DigiStrips: the user draws an horizontal “return” gesture (6th gesture of 1st row, figure 9) on the strip and the strip is enlarged (figure 11).

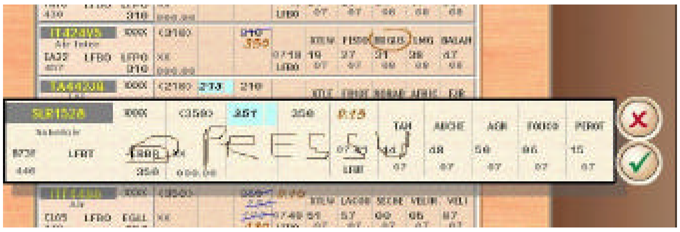


Figure 11: an enlarged strip for free-text writing.

The user may then write on the strip; nothing will be recognized. When he finished, he just push a button on the left side to close and reduce the strip to its “normal” size.

Combining graphics and animations with touch screen:

We also think that animations and graphical design is important for touch screen based interfaces. Many informal users of Virtuosi and DigiStrips expressed a strong feeling of touching something more real, less abstract, and more concrete. We are now convinced that merging these techniques dramatically enhance interfaces, but how is it possible to prove it through experiment in the ATC field?

Performance evaluation

We have compared the performance of users executing a manipulation scenario with DigiStrips on a (Cathodic Ray Tube) touch screen and with a mouse [mertz99]. We did a performance (time-to-complete) and manipulation error typology analysis (not detailed in this paper). The experiment showed a significant ($p<0.0001$) performance difference between mouse and touch screen. The latter performed 10% to 14% quicker than the mouse, even if subjects were not used to drag objects around a touch screen (but they were all "expert mouse users"). All but one (out of 8 subjects) were significantly quicker with the touch screen. The compared times include some manipulation errors corrections. Such errors were slightly more numerous with the touch screen (90 vs. 75 with the mouse) but even so, touch screens were quicker. This probably means that better trained touch screen users would probably perform even better. Finally, some touch screen manipulation errors can easily be reduced, like parallax errors (14 out of the 90 manipulation errors), with the use of flat screens.

IMPLICATIONS FOR ATC WORKSTATIONS

Beside the positive feedback obtained from air traffic controllers during the qualitative evaluations, Virtuosi and especially DigiStrips have triggered enthusiasm in the French community. However, ATC has a long history of research prototypes acclaimed by the users who evaluate it, which turn to be not so good when brought to the real world. This is probably due to the relative simplicity of the evaluations that are usually carried compared to the many levels at which a system has to be good (interaction level: usability, group work level: flexibility, task level: completeness, scalability to degraded conditions, etc.). We thus have to be very careful when analyzing the consequences of this research.

Design

The clearest lesson learnt is about the design methods and techniques used. Firstly, the DigiStrips experience confirms the efficiency of iterative design and fast prototyping, if such a confirmation were necessary. Secondly, as we already mentioned it, many aspects of careful graphical design have a positive influence on the usability and acceptability [khaslavsky] of a system. And thirdly, our informal evaluations confirm the efficiency of animations for improving feedback, and thus the usability of an interface and safety of the associated human-computer system. Though most of these findings will be hard to prove formally, we consider the current evidence sufficient for generalizing those methods and techniques.

Workstation and interaction techniques

The very positive feedback on DigiStrips and Virtuosi hint that the architecture we chose for Toccata and the interaction techniques we proposed for air traffic control are worth studying further. In particular, we are confident about the high potential of touch screens for adding future work aids to ATC workstations. However, we are very conscious that such preliminary evaluation results are not enough for us to recommend the implementation of Virtuosi or DigiStrips as they are. These applications are still incomplete, and the chosen design might prove inadequate when adding the missing functions. In addition, though strips in DigiStrips look much like paper strips, it would be inconsiderate to believe that the working methods of controllers would remain the same, and the consequences would have to be assessed. Finally, some of the questions raised by our chosen workstation architecture have to be addressed too.

To strip or not to strip?

A controversial issue with DigiStrips is that it features electronic flight strips, and this has raised a debate. The designs proposed in the past for electronic strips were not considered fully satisfactory. When Erato showed that it was possible to do without strips, there was no point debating whether electronic strips were good or not: there is no point choosing between a good apple and a bad orange. But DigiStrips reopened the debate, even between the authors of this article. Considering that there is probably not a unique solution to the problem of designing a good ATC workstation, two options can be considered:

?? For ACCs or airport towers that have no plan to move toward an Erato-like solution and that plan to use other types of working aids, a 'strip-based' evolution based on further research on DigiStrips may prove beneficial once more thoroughly evaluated. However, it should be kept in mind that, despite their good properties, strips also have intrinsic limitations in the way they support the representation of problems.

?? For ACCs which intend to use Erato-like systems, a step based on electronic strips would produce results that we cannot predict as of today. We currently have no evolution path to propose from electronic strips to a problem-based system like Erato. A promising research direction would rather be to adapt some or all of the techniques used in DigiStrips to the data representations used in Erato. Such a project is currently underway for the Erato reminder.

To be informed on DigiStrips news and follow up, please go directly to the URL: http://www.tls.cena.fr/gallery/digistrips/ or browse our Web site: http://www.tls.cena.fr/divisions/PII/ .

CONCLUSION

In this article, we have described the Virtuosi and DigiStrips prototypes of user interface for air traffic control, developed within the Toccata project at CENA. We have used those descriptions to support our arguments about the importance of professional graphic design in user interfaces for ATC, and about the possibilities of touch-screen based interaction. We have proposed a set of guidelines for design such interfaces. Finally, we have discussed the potential consequences of this research. In the future, we plan to work on devising more formal evaluations of the techniques we described, as well as on exploring more of the questions we have raised about the design of alternative workstations for air traffic control.

ACKNOWLEDGEMENT

Daniel Etienne developed Virtuosi and had bright ideas on graphic design too. He also recoded DigiStrips in perl/Tk and Zinc. Patrick Lecoanet developed the Zinc widget for the Tk toolkit, which is at the heart of the latest version of DigiStrips. Zinc will soon be available on www.openatc.org. François-Régis Colin and Michèle Jacomi developed Primo, a very first version of Virtuosi. Many people gave their time and ideas and inspired DigiStrips.

BIBLIOGRAPHY

[accot99] J. Accot, S. Zhai. Performance evaluation of input devices in trajectory-based tasks: An application of the steering law. In *Proceedings of CHI'99 Human Factors in Computing Systems*, ACM/SIGCHI, 1999.

[ahlstrom98] V. Ahlstrom, R. L. Cranston, R. Mogford, A. Raakrishnan, J. Birt, Symbol standardization in airway facilites, FAA, December 1998, DOT/FAA/CT-TN98/20.

[athenes00] S. Athènes, S. Chatty, A. Bustico. Human factors in ATC alarms and notifications design: an experimental evaluation. *Submitted to ATM'2000*.

[baber97] C. Baber, Beyond the desktop, Designing and using interaction devices. Academic Press, 1997.

[cardosi99] K. Cardosi, D. Hannon. Guidelines for the Use of Color in ATC Displays. DOT/FAA/AR-99/52. DOT-VNTSC-FAA-98-5. June 1999. National Technical Information Service, Springfield, Virginia 22161 USA.

[chatty94] S.Chatty. Issues and experience in two-handed interaction. In *Companion Proceedings of CHI94*, ACM/SIGCHI, 1994.

[chatty96] S. Chatty, P. Lecoanet, C. Mertz. Toward pen-based interaction with ATC tools. ATC Quaterly, vol. 3, 1995.

[chatty00] S. Chatty, G. Gawinowski, L. Guichard, Y. Jestin. www.openatc.org, a free software initiative for air traffic management research. *Submitted to ATM'2000*.

[faa98] Report of the computer-human interface re-evaluation of the STARS monitor and control workstation, FAA report, 1998.

[fitzmaurice97] G. Fitzmaurice W. Buxton. An Empirical Evaluation of Graspable User Interface: towards specialized, space-multiplexed input. In *Proceedings of CHI'97*, ACM/SIGCHI, 1997.

[garron92] J. Garron. Résultats des simulations Hégias. Rapport Interne CENA. CENA/92-031. Septembre 1992.

[graham97] B. Graham, Harmonisation of Human Machine Interface, the intuitive approach. In *Proceedings of the ATM'97 seminar*, 1997.

[chabrol99] C. Chabrol, J. Garron, J.C. Vigier, D. Pavet, J.L. Martin. CENA PD/3 - Final report. CENA/R99-001. February 1999.

[khaslavsky99] J. Khaslavsky, N. Shedroff, Understanding the Seductive Experience. In *Communications of the ACM*, May 1999, vol 42, n°5.

[kurtenbach95] G.Kurtenbach. Marking menus. In *Proceedings of UIST95*, ACM/SIGCHI, 1995.

[kurtenbach97] G.Kurtenbach, G. Fitzmaurice, T. Baudel & B. Buxton. The design of a GUI Paradigm based on Tablets, Two-hands, and Transparency. In *Proceedings of CHI'79 Human Factors in Computing Systems*, ACM/SIGCHI, 1997 p35-42

[kurtenbach99] G. Kurtenbach, G. Fitzmaurice, R. N. Owen, T. Baudel. The Hotbox: Efficient Access to A Large Number of Menu. In *Proceedings of CHI'99 Human Factors in Computing Systems*, ACM/SIGCHI, 1999.

[labarthe94] J.P. Labarthe. Expérimentation sur le classement de strips : résultats de la simulation. Technical report CENA/R94-022. Avril 1994.

[leroux98] M. Leroux, An in-depth evaluation of Erato tools. In *Proceedings of the ATM'98 seminar*, 1998.

[mackay98] W. Mackay, A.L. Fayard, L. Frobert, L. Médini. Reinventing the Familiar: Exploring an Augmented Reality Design Space for Air Traffic Control. In *Proceedings of CHI'98 Human Factors in Computing Systems*, ACM/SIGCHI, 1998.

[marais99] F. Marais, M. Torrent-Güell, E. Tremblay. Contraintes ERATO V1. Internal CENA report CENA/NT99775.

[maury99] S. Maury, S. Athènes, S. Chatty. Rhythmic menus: toward interaction based on rhythm. In *Proceedings of CHI'99 (late breaking results)*. ACM/SIGCHI, 1999.

[mertz97] C. Mertz. Faut-il tuer la souris? Une comparaison des charges cognitives induites par 6 procédures de saisie. In *Actes IHM'97*. Cépaduès-Editions. 1997 (p. 79-87)

[mertz98] C. Mertz, J.L. Vinot. Conception par maquettage rapide : application à des écrans tactiles pour le contrôle aérien. Dans les *actes d'ERGOIA98*. Biarritz 1998. ESTIA/ILS. pp. 120-129.

[mertz99] C. Mertz, J.L. Vinot. Touch Input Screens and Animations: more Efficient and Humanized Computer Interactions for ATC(O). In *Proceedings 10th International Symposium On Aviation Psychology*. Columbus, Ohio . May 1999.

[metcalfe98] C. Metcalfe, L. Reynolds. NATS Standard for the use of colour on air traffic control displays. R&D Report 9834. December 1998. NATS ltd, London.

[puleggi98] F. Puleggi. Vers des méthodes d'interaction répartie: étude expérimentale et premières analyses théoriques. *Masters thesis, ENAC*. Toulouse, 1998.

[shneiderman93]. *Sparks of Innovation in Human-Computer Interaction*, B. Shneiderman editor, Ablex Publ, NJ (1993), 2nd edition.

[wellner93] P. Wellner, W. Mackay, R. Gold. Computer Augmented Environments. *Special Issue of Communications of the ACM*, 23(7), Juillet 1993.

BIOGRAPHIES

Christophe Mertz works at Centre d'Etudes de la Navigation Aérienne (CENA) on hire from the French software house CR2A-DI. He graduated at Ecole Nationale Supérieure de Télécommunication de Bretagne (ENSTBr, Brest, France). From 1991 on he has led research projects on Human Computer Interfaces applied to Air Traffic Control. He currently explores uses of touch screens, animations and graphic design for ATC workstations. He co-designed DigiStrips with Jean-Luc Vinot.

Stéphane Chatty is head of the "Environment of en-route controllers" department at CENA. He graduated at Ecole Polytechnique (Paris, France) and Ecole Nationale de l'Aviation Civile (Toulouse, France), and holds a PhD in computer science from University of Paris-Sud.

Jean-Luc Vinot is a user interface (UI) designer at CENA. He graduated at Ecole Nationale Supérieure d'Arts Appliqués et Métiers d'Arts (ENSAAMA) and specialized in graphic design and typography. He has worked since 1995 in a human computer interaction research group at CENA. His interests are the contribution of graphic art to User Interfaces design and how to humanize User Interfaces through better graphic design and new interactions. He designed and coded DigiStrips.