

Gesture Wars - jnd.org

At the start of almost every technology transition, chaos rules. Competitors create confusion, often quite deliberate, as they develop their own unique way of doing things incompatible with all others.

A challenge is arising as gesture-based control takes over on cellphones, tablets, touchpads and computers. Change invariably creates confusion and this situation is exacerbated by the different design philosophies of competing companies coupled with the lack of standardization. This problem is compounded because the new modes of interaction ignore the many important lessons of proper interface design, including discoverability, feedback and the power of "undo."

Today, the long-established, well-learned model of scrolling is being changed by one vendor, but not by others. Gestures proliferate, with no standards, no easy way of being reminded of them, no easy way to learn. Change is important, for it is how we make progress. Some confusion is to be expected. But many of these changes and the resulting confusions of today seem arbitrary and capricious.

The Great Debate Over the User's Model of Scrolling

Back in the early days of computer displays, a great fight ensued over the correct user model for scrolling. Consider the simple paradigmatic case of material on a screen where not all can fit within the available space. The bottom of the visible window is not the end of the material. Imagine that the material is actually located on a long vertical roll with the only part visible being that which is visible through the window. To see material not visible, there are two choices: move the material or move the window. If the material is being moved, then scrolling up moves the material up. But if it is the window that is moved, then scrolling down makes the text appear to move up.

Both models are correct in the sense that both make logical sense. The "correct" answer is that the method of scrolling should match the user's conceptual model of the activity (usually called the user's mental model). Whichever method is adopted then requires that all people learn to see the world through that particular conceptual model.

A very similar debate took place in the aviation community about the proper way to display an airplane's attitude on the graphical display in the cockpit. This debate is highly relevant here, because when a new technology emerged, it changed the nature of the debate. The display showed the silhouette (seen from directly behind the airplane) superimposed over a horizontal line that represented the horizon. The question being debated was an exact analog of the question in the computer world.

If the airplane was banking left, what should the display look like: outward-in or inward-out? The outward-in user model shows what an observer placed directly behind the airplane would see. In this case, the display would show the horizon remaining stable (horizontal) and the airplane tilting to the left. The inward-out user model shows the view seen by the pilot looking out the window of the airplane. In

this case, the display would show the airplane remaining stable (upright) and the horizon tilting to the right. Which is correct? Both are. Which was adopted? Both. That is, the fight was unresolvable, both in the academic journals and in the real world of pilots and airplanes. The result was what you might expect: an acceptable amount of confusion. Pilots who were certified to fly a particular airplane (for example, a Boeing 737 that used the outside-in displays) had to be retrained when switching to a 737 that used the inside-out displays. But, overall, allowing each airline to select the preferred choice of instrumentation worked well.

When the heads-up display (HUD) entered the world of commercial aviation, the nature of the debate changed. A HUD is a simplified set of critical displays projected in space out the windshield, so the pilot views it while looking out the windshield. When a pilot flies by looking out of the window (at the real earth as well as at the HUD), the HUD has no choice but to use the inside-out format. (The debate about the attitude display is actually rather lengthy and technical, and because this column is about a different topic, I'll stop here.)

With window displays in computer operating systems and applications, the world converged on scrollbars as the control mechanism—the "move the window" paradigm for control. This means that all of us, no matter what operating system we use—for example, Windows, Macintosh or Unix/Linux—move the scrollbar down in order to move the material up.

There was always one interesting exception. Many graphics programs allowed the mouse to "grab" the material displayed and move it directly. This mode was usually indicated by two factors: the mouse cursor was placed on the item to be moved rather than on a scroll bar and the cursor changed into a hand, sometimes shown clenched so as to indicate "grabbing" the material to be moved. In this case, one moved the mouse up in order to move the material being displayed up.

The Scrolling Model Changes

The emergence of multi-touch screens on phones, tablets and computer screens change the interaction model—with a touch screen, one directly manipulates the material, not the scrollbar. Now it makes sense to use the "hand" model, where touching the image of the items on the screen and dragging or flicking it upwards moves the material upward, no scrollbar being required.

OK, that is sensible. When touching the screen image, whether metaphorically (as in the case of drawing programs and the image of a hand) or literally, as in gestural control of touch-sensitive screens, move in the direction one wants the image to move. When using a scroll bar, move the bar down to move the window up, and the display down. So what is the problem?

The problem occurs when the neat logical distinction between moving a scrollbar versus moving the displayed material confuses the average user. Moreover, small displays such as those found on cellphones and tablets tend not to use windows. When a single application occupies the entire display, the scrollbar is unnecessary.

When Apple first introduced the trackpad on its portable computers and later, on its touch-sensitive mouse and separate trackpad, it followed the standard computer model: dragging two fingers down the touch surface moved the window down and the displayed material up.

Apple has now decided that the discrepancy between the scrolling model for scrollbars and gestures should be eliminated. Computers mostly still lack touchscreen interfaces, especially multi-touch, and they still use windows and scrollbars. Nonetheless, in Apple's latest version of its operating system (OS 10.7, otherwise known as Lion), the default model has been changed: one moves the material up, not the window down. Apple wants everyone to move the material with a two-finger gesture, moving the two-fingers down the screen (on a touch screen) or on a trackpad. Yes, there still is a scrollbar that still seems to use the old mode, but I predict that scrollbars will disappear as control devices. Indeed, in new applications the scroll bar is hidden, only becoming visible when the two-finger scroll is initiated. Although it can be grabbed and moved, the scrollbar's main function now is to indicate what part of the material is visible through the window.

The result has been great confusion among customers. Suddenly, the well-ingrained habit has been reversed. Apple has long had a touchpad on its portable machines as well as being sold as an external control device. But the two-fingered drag downward used to move the material upward—that is, it controlled the scrollbar. Now the same movement controls the material displayed, so moving downward moves the material displayed downward.

The change extends to the way that the center control wheel on the mouse moves the material on the screen, but not to cursor arrows or the "page up" and "Page down" keys on the keyboard. "Page down" moves the text up, as does depressing the down arrow key once the cursor has reached the bottom of the screen.

The reason for the change is, presumably, consistency. Gestures are becoming the standard way of moving material around on multi-touch screens and multi-touch will become standard on all systems in the next few years—either through touch screens or touchpads (or more likely, both).

Do we need consistency? And what do we mean by consistency, anyway? In Apple's case, I can only assume that they thought consistency is measured at the level of the hand: move hand up, material moves up. But as long ago as 1995, researchers showed that in certain situations people preferred a mixed model, where in some cases the hand moved with the document, in others against it. Consistency should be measured at the level of the mental operation. If it seems natural to move the material being viewed, then hand motion should be in accordance with document motion, but if it seems more natural to move the window, then hand motion should move in accord with the window movement. In other words, although the hand movement might seem inconsistent at the physical level of the hands and document, it can be completely consistent with the person's mental model because to the person, different items—document or window—are being moved in the two cases, and the object being moved does follow the hand motion. Consistency, therefore, has to be measured in the mind, not the world.

(See Frederiksen. et al.)

But scientific, academic arguments aside, what does it mean for Apple to have changed the rules of the game? My prediction is that although it will cause great confusion and uproar among Apple's customers because their long-tuned habits have been violated, in fact, it is not that difficult to change the mental model from moving the window to that of moving the material. People will find that in a few hours, perhaps a few days, it all will seem natural again.

The Confusion has Just Begun

But actually, the confusion has just begun. Microsoft faces the same issue about the scrolling model as it deploys gesture systems on everything from its "surface" product, to smart phones, tablet computers, regular computers and its touch mouse. Which model will they adopt? So far, Microsoft is sticking with the current model of moving the window: move the finger up to scroll the material down.

Both Microsoft and Apple can make good arguments for selecting either decision. The problem is that if the two dominant companies make different decisions, chaos and confusion will result. Anyone who is monolingual and only uses a Mac or a Windows machine will be OK: Those of us who are bilingual will have problems. When I use a mouse on my Apple Macintosh, scrolling the center wheel upward moves the text upward. When I use a very similar mouse on my Microsoft Windows computer, scrolling the center wheel upward moves the text downward. And because Apple computers can run both Macintosh and Windows operating systems simultaneously, this discrepancy can happen with the identical physical equipment. Of course it will be confusing.

The story gets even worse. First, many other companies are deploying gesture based display devices: which model will they follow? Second, the problem is not restricted to scrolling. Each company allows a wide range of gestures, so the number of gestures that one has to remember is already in the double digits and ever increasing and, of course, the two major platforms are quite incompatible.

Each new vendor will have its own design philosophy, further compounding the problem. For example, Google supports gestures with its Android platform and Chrome web computer, but differently than either Microsoft or Apple, following their own design guidelines. Not only does the screen material move in different directions, but for Microsoft, scrolling only requires one finger on the mouse, while for Apple, it takes two fingers. The differences do not stop there. The platforms allow a wide variety of gestures:

- One finger can be used to touch, tap, or double tap. One can tap and hold or rotate clockwise or counterclockwise. It can also swipe up, down, left or right. Microsoft allows the thumb to have a special meaning, something that Apple has not yet done. Google uses a long press to call up a menu, although this is seldom actually used by their developers or not even by Google itself.
- Two fingers can be used to in the same ways as one, as well as to pinch or spread. With Microsoft's mouse, the location of the tap matters. I suppose you could have a two-finger long press, although to my knowledge nobody yet uses this.

- Three and four fingers can be used in a variety of ways, some involving the thumb.
- Some gestures involve movement of the device, using the location orientation, and acceleration sensors present in portable devices. Some gestures involve tapping the case or blowing across the microphone. Some involve tilting, tapping, or rapid shaking of the entire device.
- With video cameras watching the user, such as with Microsoft's Kinect, the gestures can be made in three dimensions without contacting anything using fingers, hands, feet, the whole body, or just the head.

Not only is there incompatibility among the vendors, but given the lack of any cues on the devices, it is very difficult to remember the gestures. We are back to the days of command-line interfaces where everything had to be memorized, or looked up in a manual. I can only remember the gestures on my Macintosh by launching the "Settings" application, finding the trackpad control panel, and reviewing the four gestures under the tab "Point & Click," the four gestures under "Scroll & Zoom," and the six gestures listed under "More Gestures." What ever happened to Apple's image of "ease of use?" In the early days, one didn't even have to read the manual to use a Macintosh. Now I am forced to read the same section of the manual on a regular basis.

Just today I read in *The New York Times* that some actions in some applications on the Apple iPhone or iPad can be undone by shaking the device: "hold the iPad firmly with both hands (please)" warns *The New York Times*. Who would ever have discovered this without the newspaper?

Where will it all lead? Incredible excitement coupled with incredible confusion. In the end, we are going to need standard sets of gestures. Will this happen? Oh yes, but the question is when. Today, everyone is patent happy and every small little action is patented, so Microsoft will have its proprietary set, Apple and Google will have theirs, all the other vendors will have theirs. Each vendor has a different philosophy of interaction. Thus, Google's Android encourages the use of menus and the "back" key, even providing permanent buttons on the device for these purposes. Apple dislikes menus (although they helped pioneer their use on computers). But without menus, the set of actions one can take are invisible and as a result, almost undiscoverable. Couple this to the lack of consistency, the lack of visible design cues as to the possible actions, and the power and excitement of gesture-based systems will be replaced by chaos and confusion.

Jakob Nielsen and I have explored some of these issues in separate articles and one joint article (see the list in the notes at the end of this column). Readers somehow seem to think that we are enemies of gesture-based systems, but these readers have either misread us or, more likely, criticized without reading. We are fans of progress, but enemies of confusion. As this article points out, some confusion is to be expected when old habits must be retrained, but much of the confusion today represents inappropriate business and marketing decisions rather than design decisions. The customer is ill served.

We are in a period of exciting changes. The confusions that result will eventually dissipate, but in the meantime, nobody is well served.

Aside on the History of Gestures and Scrolling Models.

Originally, I intended to include a brief history of the development of the scrolling model. I tried to trace the history of the debate about text movement by emailing my friends who were directly involved in making these decisions. Each of them thought one of the others had made the decision.

I was able to trace the debate back to the early 1970s and even to the cursor keys on the old text-based video display terminals (VDTs) of the 1960's. Many people claim the convention started at Xerox PARC, but my contacts who were there during the development of the first graphical user interfaces (Alto and Star) say the standard probably developed earlier and they followed it. There were many debates about the arrows on scroll bars: where should they be placed; in which direction should they point?

The debate continued in the academic journals for a long time: the last paper I encountered was in 1995, but there are probably later ones. One paper pointed out that both schemes could be found in the Visual Display Terminals being used in 1982, and that the two different schemes could be found "even within the product line of one manufacturer" (Bury, et al.).

In the end, the history was both fascinating, rich with stories and details, but far longer than was reasonable for this column. So it is now in my "to be written" list.

Thanks

Special thanks to the many people who responded to my request for assistance in tracing the history of scrolling models, too numerous to list here, but it includes some of the great names in the early history of personal computing.

References

Bury, K. F., Boyle, J. M., Evey, R. J., & Neal, A. S. (1982). Windowing vs. scrolling on a visual display terminal. *Proceedings of the 1982 conference on Human factors in computing systems*.

Frederiksen, N., Grudin, J., and Laursen, B., 1995. Inseparability of design and use: An experimental study of design consistency. *Proc. Computers in Context' 95*, 83-89.

Norman, D. A. (2010). Natural User Interfaces Are Not Natural. *Interactions*, 17, No. 3 (May - June).
http://www.jnd.org/dn.mss/natural_user_interfaces_are_not_natural.html

Norman, D. A. & Nielsen, J/ (2010). Gestural interfaces: a step backward in usability. *Interactions* 17, No. 5 (September - October), 46-49.
http://www.jnd.org/dn.mss/gestural_interfaces_a_step_backwards_in_usability_6.html