

# Jensen Harris: An Office User Interface Blog

## Giving You Fitts

22 Aug 2006 10:00 AM

One of the most well-understood and salient principles underlying the ergonomics of graphical user interface design is [Fitts' Law](#).

Named for Paul Fitts, a psychologist at Ohio State University, Fitts' Law is a mathematical model of fine motor control which predicts how long it takes to move from one position to another as a function of the distance to and size of the target area. Papers outlining what became known as Fitts' Law were published in 1954 and 1964.

Fitts himself was an expert in aviation psychology, and he developed his research around more ergonomic layouts for cockpit instrumentation as a way of increasing aviation safety. You can read more about the [early history and mathematics behind Fitts' Law on Wikipedia](#).

Fitts' model proved especially relevant to the early research on computer input devices performed in the late 1970s. Although Fitts' model was originally formulated to project how quickly a human could point at a physical button, it turns out that the same set of rules governs how quickly someone can target an area on the screen with a mouse cursor.

Although there's a great deal of subtlety to Fitts' research, what became known as Fitts' law is a fairly simple intuitive concept.

1. The farther away a target is, the longer it takes to acquire it with the mouse.
2. The smaller a target is, the longer it takes to acquire it with the mouse.

The inverse of both statements is true as well (closer and bigger targets can be more quickly acquired.)

One common mathematical formulation of this relationship is:

$$MT = a + b \log_2 \left( \frac{2A}{W} + c \right)$$

- $MT$  is the average time taken to acquire the target.
- $a$  and  $b$  are empirical constants determined through linear regression.
- $A$  is the distance from the starting point to the center of the target.
- $W$  is the width of the target measured along the axis of motion (how close to the target you need to get to count as acquiring it.)
- $c$  is a constant which is either 0, .5, or 1, depending on the specific environment.

Here's a cool Java-based applet which lets you play around with Fitts' Law to see how it feels in practice: <http://www.tele-actor.net/fitts/>

## How Fitts' Law Affects User Interface

The key takeaway for interface designers is clear: the farther away a button is from the current mouse position, the larger it needs to be to achieve the same average acquisition speed. Put another way, there are two main ways to improve mouse efficiency: put the controls closer, or make them bigger. *(There are other more avant-garde ways to alter the physics of mouse travel which I won't go into today.)*

Over the years, as monitors have gotten bigger and screen resolutions have increased, Fitts' law dictates that actual mouse efficiency has gone down.

Think about Word 1.0, which was designed for a common maximum 640x480 screen resolution. Toolbar buttons in Word 1.0 were 20x20 buttons with 16x16 icons in them.

Word 2003, on the other hand, is commonly run at resolutions as high as 1600x1200 and beyond--yet the toolbar buttons remain the same 20x20 size they were in Word 1.0. But because the screen is so much larger, most of the time your mouse cursor will be much farther away than it could have been on a 640x480 screen. Greater mouse distances mean an increased *MT* target acquisition time.

In other words, the same button takes much longer to click than it did fifteen years ago.

## Mile-High Menus and Magic Corners

One of the most useful aspects of applying Fitts' Law to computers is that screen size is bounded. No matter how far you move your mouse to the left, the cursor will never go farther than the left side of the screen.

In a Fitts' Law sense, you can think of the edges of the screen as being infinitely wide.

Think about how long it would take you to move your cursor from the right side of the screen to the left edge of the screen. Now compare that to how long it would take you to move your cursor from the right side of the screen to a spot 2 pixels from the left edge.

Obviously, you would be much faster in the first case because you can literally slam your mouse to the left as fast and as hard as you want and you won't overshoot. It's infinitely wide.

This is the same reason that the Mac user interface has been said to have "mile-high menus." The Mac menu bar is permanently affixed to the top of the screen regardless of what program you're in. As a result, you only have to worry about targeting a menu horizontally. Because the top edge of the screen is essentially "infinitely" tall, you can acquire the menus very quickly.

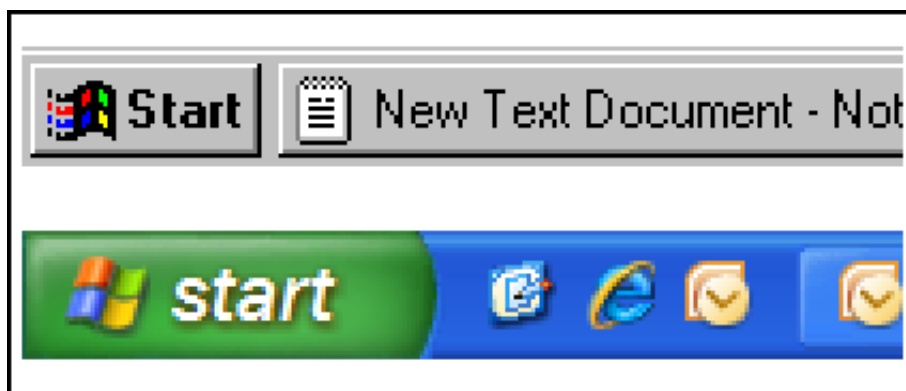
The Windows taskbar is a mile-high the other direction: you can move your mouse to the bottom of the screen quickly and only worry about targeting horizontally. *(If you resize your taskbar to be two rows high, of course, all bets are off.)*

Wait, it gets even better. There are four places on the screen that are effectively both infinitely wide and infinitely tall. You guessed it: the four corners.

Regardless of how distant a corner is from your current mouse position, you can get to the corner in no time at all. Acquiring the corner requires very little fine motor control at all because the virtual target is so huge. In GUI terms, the corners are so good they're often called "magic."

The Start button in Windows is seemingly located in an ideal place for fast acquisition, and in recent

versions of Windows that's certainly true. Prior to Windows 2000, however, the Start button had a single "dead" pixel along the left and bottom sides of it in which clicking didn't open the Start menu. The result: slower acquisition times and a startling number of missed clicks.



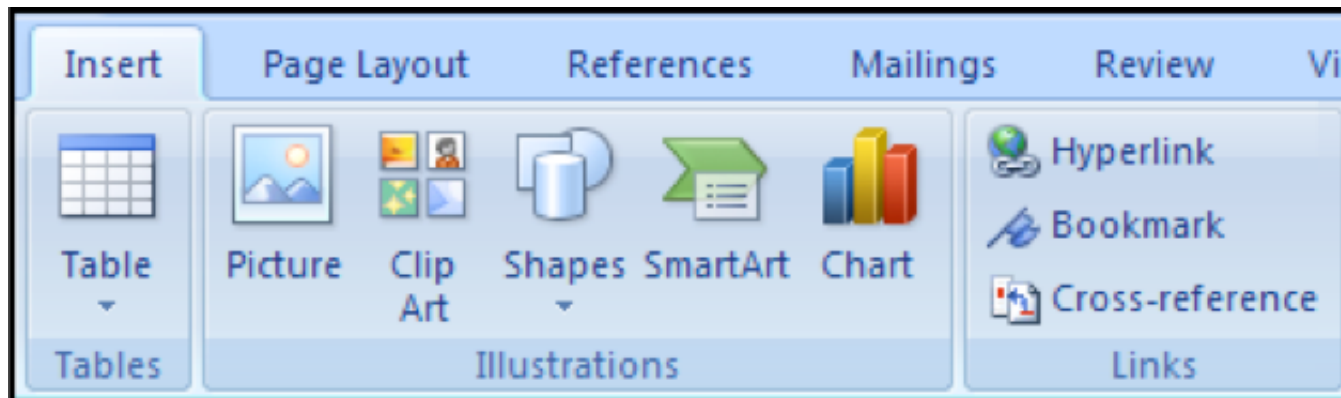
*Windows 95: Missed by a pixel  
Windows XP: Good to the last drop*

Happily, the Windows team fixed this almost eight years ago.

### Office 2007 and Fitts' Law

We've tried to pay attention to Fitts' Law throughout the redesign of the Office user interface.

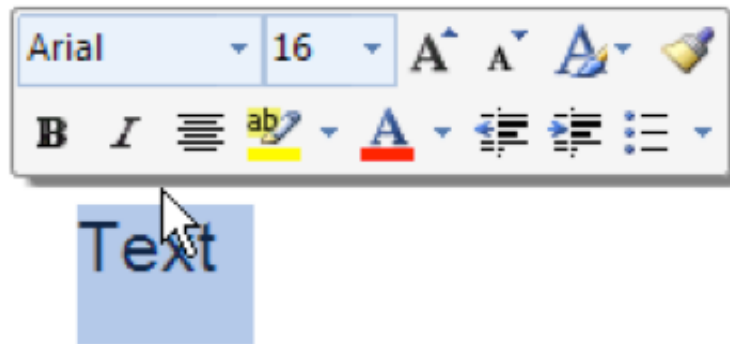
First off, most controls in the Ribbon are labeled. This helps discoverability and usability considerably, but it also makes the buttons bigger and easier to target. As your screen resolution increases, the width of the Ribbon also increases, providing room for more labels and larger buttons.



*Larger, labeled controls can be clicked more quickly*

In a sense, the Ribbon tries to keep  $MT$  from Fitts' Law relatively constant by compensating for the greater average travel distance required at higher resolutions by displaying larger controls.

The [Mini Toolbar](#) was designed with Fitts' Law in mind as well. Whenever you select text or right-click selected text, a small toolbar appears directly next to the mouse cursor (*you've seen the movie, right?*) As you move closer to it, it fades in; as you move away, it fades out.



*Mini Toolbar: Close to the cursor*

The controls on the Mini Toolbar are small, but because they're located directly next to your cursor, they're easy to target. In this case, you want to have small buttons because it means you can have as many as possible located as close as possible to the cursor.

We did look at other more radical designs (such as positioning the Mini Toolbar directly on the cursor, or a radial design) but we were also trading off with being able to see the text you've just selected and how easy it is to scan the controls on the toolbar. The design we went with provided the best overall balance of efficiency and utility.

The Ribbon is designed to increase *W*; the Mini Toolbar is designed to reduce *A*. Both of these affordances help to reduce *MT*, the time it takes to click a button.

### Quick Access a Mile High

The operating system really has the best opportunity to take advantage of the edges and corners of the screen. When Office windows are floating around on the desktop, we're sort of confined to the window we're in.

But there's good news: according to the [Customer Experience Improvement Program data](#), a startling number of Office windows run maximized. Even at high resolutions like 1280x1024 and 1600x1200, Office windows are maximized most of the time. And at 1024x768 and below, we're maximized almost all the time.

Why is this good news? Because when we're maximized, we suddenly get edges of the screen to play with. The right edge of the screen is used for the scroll bar, which we are careful to make sure extends all the way to the edge of the screen so that it's a "mile wide." The left edge of the screen is used differently in each program.

Historically, the top edge of the screen is used for the title bar. Having a title bar is probably necessary, but it's a huge waste of easily-targeted space, especially when your window is maximized (meaning that you're not dragging it around to move it anyway!)

So, we wanted to take advantage of the title bar space to help make certain controls faster to target; this is why the [Quick Access Toolbar](#) is located in the title bar by default.

We designed the customizable Quick Access Toolbar to contain features people use frequently and regardless of the Ribbon tab they're on. By default, it contains Save, Undo, and Redo, but you can add any control in the Ribbon to your QAT by right-clicking it and choosing "Add to Quick Access Toolbar."

Because the Quick Access Toolbar is in the title bar, the buttons are effectively infinitely tall. You can

target and click each of the buttons very quickly; they're a "mile high." Finally, you can reclaim this valuable screen edge to put features you want to access ultra-efficiently.



*Quick Access that's a mile high...*

*(Note: In Beta 2, there's a bug which keeps these controls from extending to the top of the screen; it's fixed in the upcoming Beta 2 Technical Refresh.)*

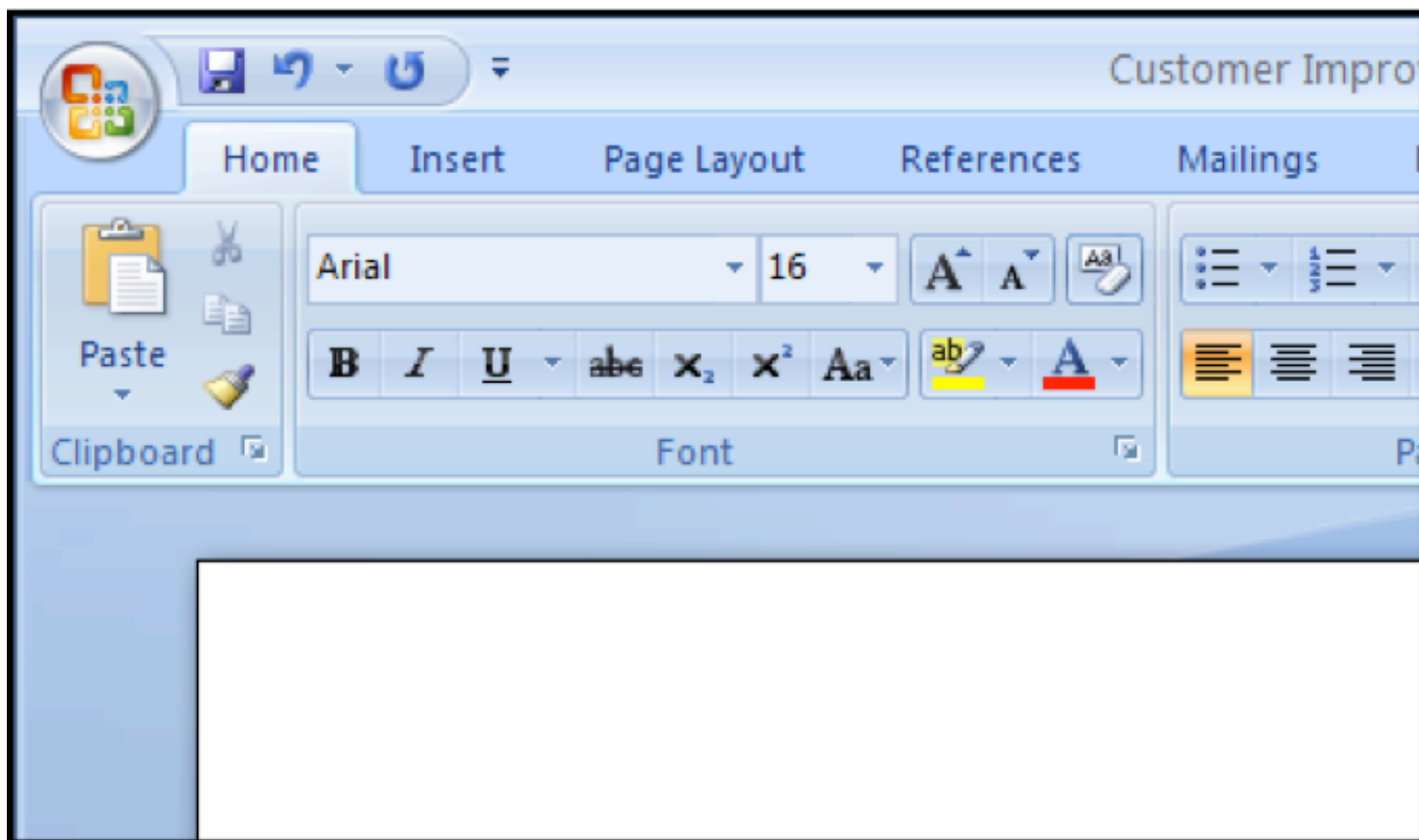
### **And We Didn't Leave Out the Magic Either...**

I mentioned before how special the corners of the screen are because they're effectively infinitely tall and wide.

The bottom-left and bottom-right corners are taken up by the Windows taskbar, so those can't be used by Office. The upper-right corner is used for the window close button in each app, so it's kind of off-limits as well.

The upper-left corner, though, in most programs is used for a system menu which is mostly intended to be used via the keyboard: not a good use of the most premium real-estate on the screen.

In Office 2007, we decided to take advantage of that corner by using it for the Office button.



*Magic Corner: Office Button in the upper-left corner*

Although the button itself is round, the hit target for it actually extends on a maximized window all the way to the upper-left edge of the screen.

As a result, accessing the Office menu to Save, Open, Print, Send (or to access one of your Recent Documents) is ultra-efficient. Fitts' Law was actually one of the driving forces behind the Office Button design.

## Summary

Speed of target acquisition is but one of the many characteristics of a graphical user interface, but it's an important one. In Office's redesign, we've tried to take advantage of Fitts' Law in several key ways: the control layout and scaling of the Ribbon, the Mini Toolbar and other "by the cursor" contextual UI, and the usage of the edges and corners of the screen for the Quick Access Toolbar and Office Button.