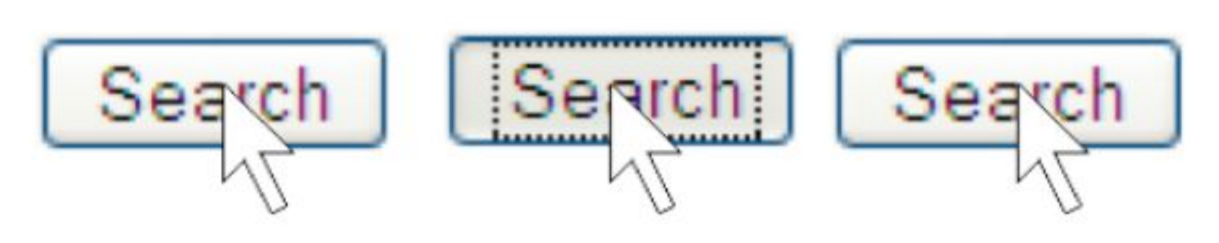
Feedback

Actions Should Have Immediately Visible Effects

- Low-level feedback
  - e.g. push button



- High-level feedback
  - system state changes
  - new web page starts loading

- Hand-in-hand with affordances is feedback
  - how the system changes visibly when you perform an action.
- When the user invokes a part of the interface, it should appear to respond.
  - Push buttons should depress and release.
  - Scrollbar thumbs and dragged objects should move with the mouse cursor.
  - Pressing a key should make a character appear in a textbox.
- Low-level feedback is provided by a view object itself, like push-button feedback.
  - This kind of feedback shows that the interface at least took notice of the user’s input, and is responding to it. (It also distinguishes between disabled widgets, which don’t respond at all.)
- High-level feedback is the actual result of the user’s action, like changing the state of the model.

Perceptual Fusion

- Two stimuli within the same perceptual cycle (T_p ~ 100ms [50-200 ms]) appear fused
- Consequences
  - 1/ T_p frames/sec is enough to perceive a moving picture (10 fps OK, 20 fps smooth)
  - Computer response < T_p feels instantaneous
  - Causality is strongly influenced by fusion

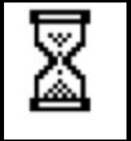
- One interesting effect of human perceptual system is perceptual fusion. Here’s an intuition for how fusion works.
- Our “perceptual processor” runs at a certain frame rate, grabbing one frame (or picture) every cycle, where each cycle takes T_p seconds.
- Two events occurring less than the cycle time apart are likely to appear in the same frame.
- If the events are similar – e.g., Mickey Mouse appearing in one position, and then a short time later in another position – then the events tend to fuse into a single perceived event – a single Mickey Mouse, in motion.
- The cycle time of the perceptual processor can be derived from a variety of psychological experiments over decades of research
    - summarized in Card, Moran, Newell, The Psychology of Human-Computer Interaction, Lawrence Erlbaum Associates, 1983
- 100 milliseconds is a typical value which is useful for a rule of thumb.

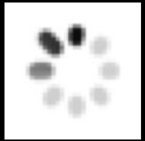
- But it can range from 50 ms to 200 ms, depending on
  - the individual: some people are faster than others
  - the stimulus: for example, brighter stimuli are easier to perceive, so the processor runs faster
- Perceptual fusion is responsible for the way we perceive a sequence of movie frames as a moving picture, so the parameters of the perceptual processor give us a lower bound on the frame rate for believable animation.
  - 10 frames per second is good enough for a typical case, but 20 frames per second is better for most users and most conditions.

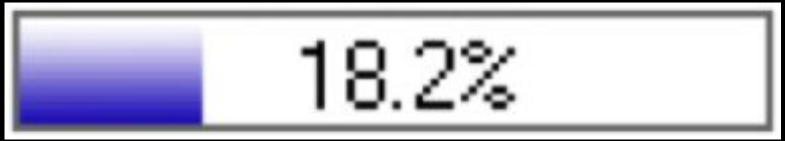
- Perceptual fusion also gives an upper bound on good computer response time.
- If a computer responds to a user’s action within T_p time, its response feels instantaneous with the action itself.
  - Systems with that kind of response time tend to feel like extensions of the user’s body.
  - If you used a text editor that took longer than T_p response time to display each keystroke, you would notice.
- Fusion also strongly affects our perception of causality.
  - If one event is closely followed by another – e.g., pressing a key and seeing a change in the screen – and the interval separating the events is less than T_p, then we are more inclined to believe that the first event caused the second.

Response Time

- < 0.1 s: seems instantaneous
- 0.1-1 s: user notices the delay
- 1-5 s: display busy indicator
- 1-5 s: display progress bar







Feedback Visibility Depends on Locus of Attention

- Attention focuses on one input channel (e.g. area of visual field) at a time
- Does the user’s locus of attention include:
    - Caps Lock light on keyboard?
    - Status bar?
    - Menu bar?
    - Mouse cursor?

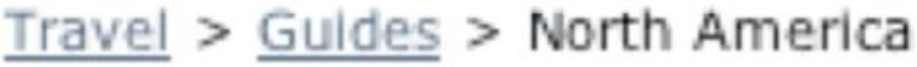
- The metaphor used by cognitive psychologists for how attention behaves in perception is the spotlight:
    - you can focus your attention on only one input channel in your environment at a time.
    - This input channel might be a location in your visual field, or it might be a location or voice in your auditory field.
    - You can shift your attention to another channel, but at the cost of giving up your previous focus.

- So when you’re thinking about how to make something important visible, you should think about where the user’s attention is likely to be focused
  - their document? The text cursor? The animated banner ads on the web site?
- Raskin (The Humane Interface, 2000) has a good discussion of attention as it relates to mode visibility.
  - Raskin argues that we should think of it as the locus of attention, rather than focus, to emphasize that it’s merely the place where the user’s attention happens to be, and doesn’t necessarily reflect any conscious focusing process on the user’s part.

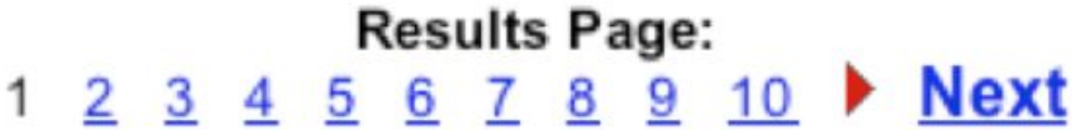
- The status bar probably isn’t often in the locus of attention.
    - There’s an amusing story (possibly urban legend) about a user study mainly involving ordinary spreadsheet editing tasks, in which every five minutes the status bar would display “There’s a $50 bill taped under your chair. Take it!” In a full day of testing, more than a dozen users, nobody took the money. (Alan Cooper, The Inmates Are Running the Asylum.)
- But there’s also evidence that many users pay no attention to the status bar when they’re considering whether to click on a hyperlink; in other words, the URL displayed in the status bar plays little or no role in the link’s information scent (which we’ll discuss next).
- Phishing web sites exploit this to hide their stinky links.

Visible Navigation State

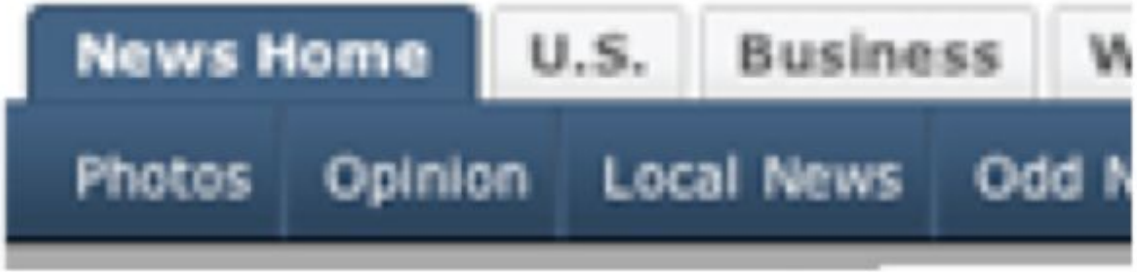
- Breadcrumbs



- Pagination



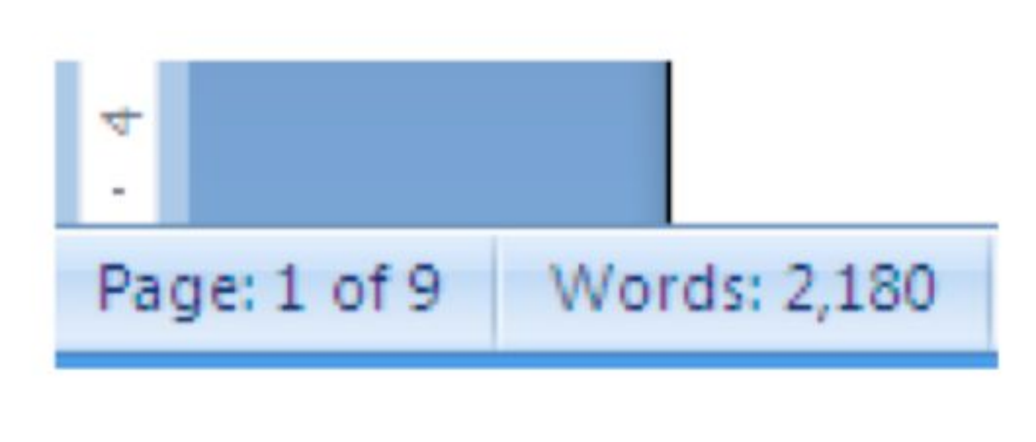
- Tabs



- Navigation is one important kind of state to visualize
    - i.e., where am I now?
- On the Web, in particular, users are in danger of getting lost as they move around in deep, information-rich sites.
- Breadcrumb trails show where you are as a path through the site's hierarchy (e.g. Travel, Guides, North America), in a very compact form.
- Showing the hierarchy in a tree widget with the current node highlighted is another way to do it, but costs more screen space and complexity.
- Pagination and highlighted tabs are similar patterns that show the user where they are, along with some context of where else they could go.

Visible Model State

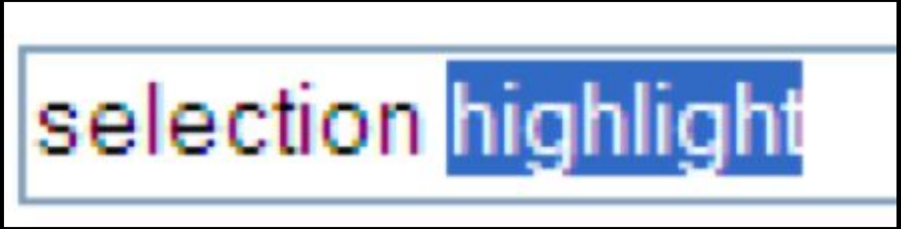
- Continuous visual representation of model
  - What to visualize should be guided by the user's tasks

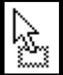
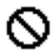


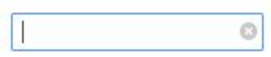
Page: 1 of 9 | Words: 2,180

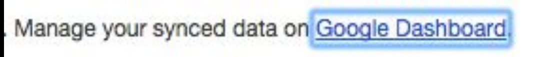
- The hard design issues in model visibility tend to lie in what to make visible (i.e. which aspects of the model), and how to display it (i.e., in what representation).
- The what may involve a tension between visibility and simplicity; visibility argues for showing more, but simplicity argues for showing less.
- Understanding the users and their tasks (a technique called task analysis) helps resolve the tension.
- For example, Microsoft Word displays a word count continuously in the status bar, since counting words is an important subtask for many users of Word (such as students, journalists, and book authors).
- Making it always visible saves the need to invoke a word-count command.

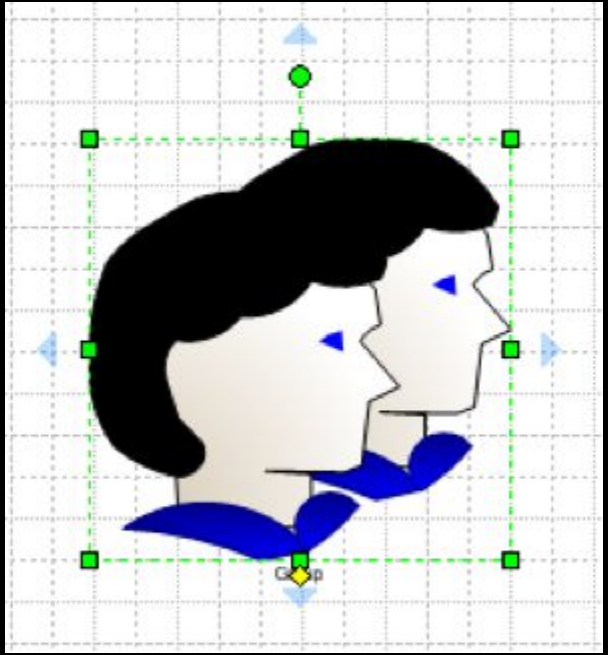
Visible View State


selection highlight

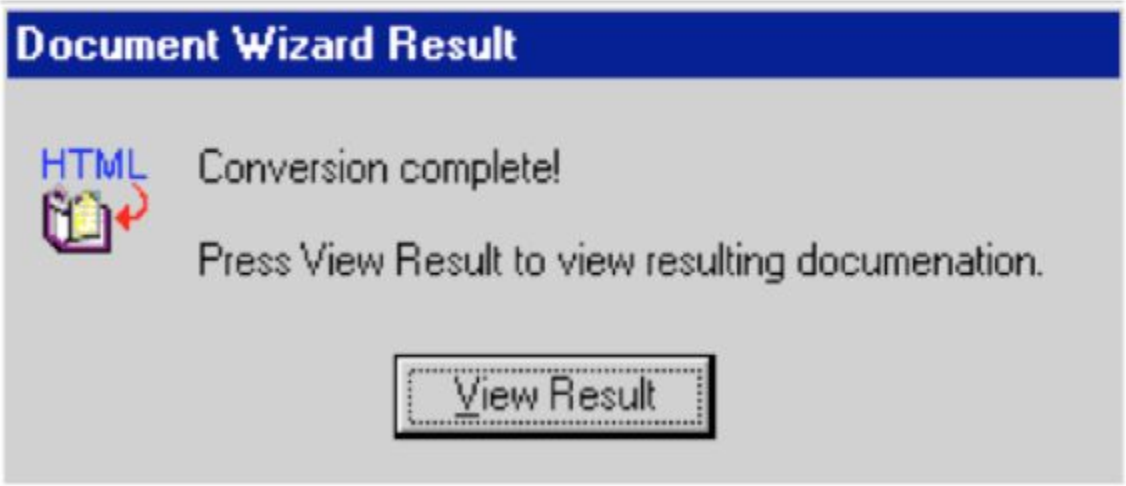
- Selection highlight
- Selection handles
- Drag & drop mouse cursor
  - dragging 
  - can't drop 
- Keyboard focus




Manage your synced data on Google Dashboard.



Useless vs. Useful Feedback



Document Wizard Result

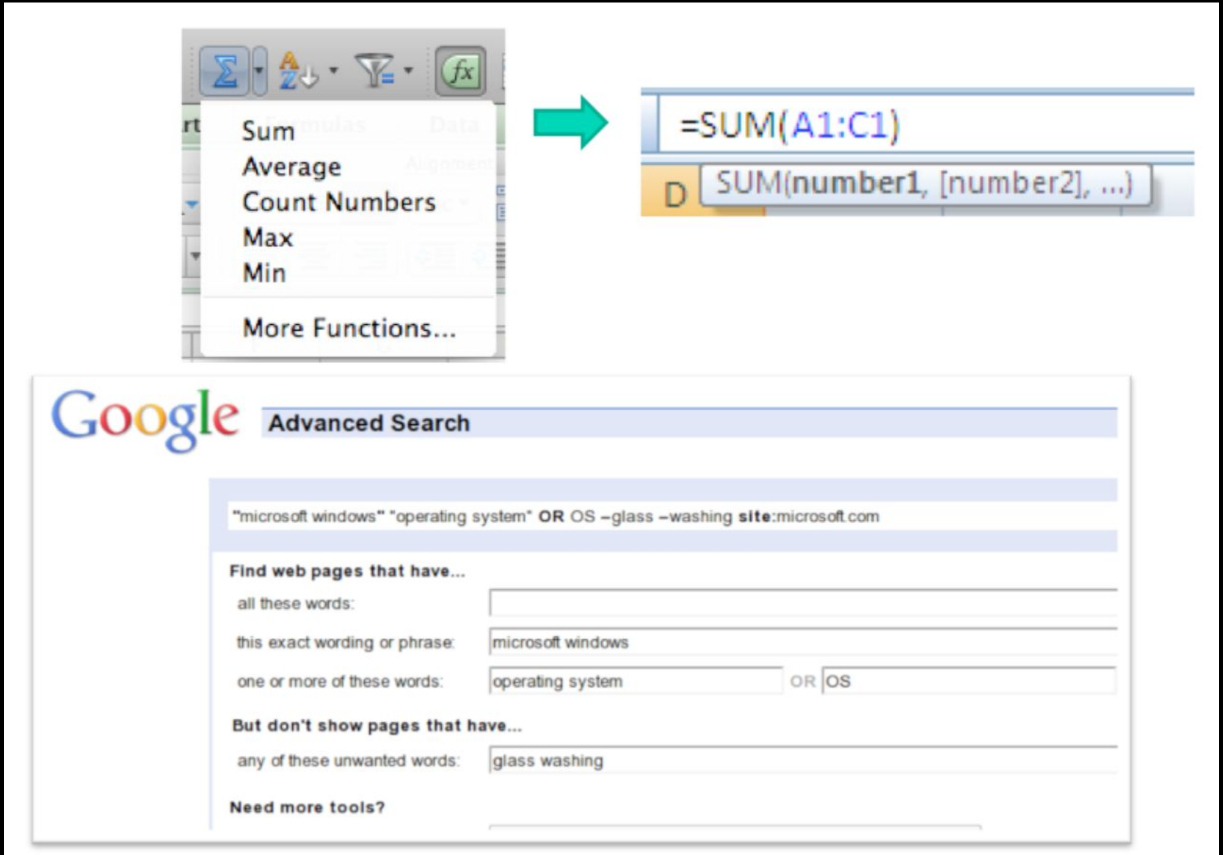
Conversion complete!

Press View Result to view resulting documenation.

Source: Interface Hall of Shame

Feedback is important, but don’t overdo it. This dialog box demands a click from the user. Why? Does the interface need a pat on the back for finishing the conversion? It would be better to just skip on and show the resulting documentation.

Self Disclosure: Teaching through Feedback



=SUM(A1:C1)

SUM(number1, [number2], ...)

Advanced Search

"microsoft windows" "operating system" **OR** OS –glass –washing **site**:microsoft.com

Find web pages that have...

all these words:

this exact wording or phrase: microsoft windows

one or more of these words: operating system OR OS

But don't show pages that have...

any of these unwanted words: glass washing

Need more tools?

Self-disclosure is a technique for making a command language more visible, helping the user learn the available commands and syntax.