Abram Hindle
Department of Computing Science
University of Alberta

Human Error

Human Error

Goal:

- could the designer have improved the user interface to reduce the chance of human error?

Reducing Human Errors

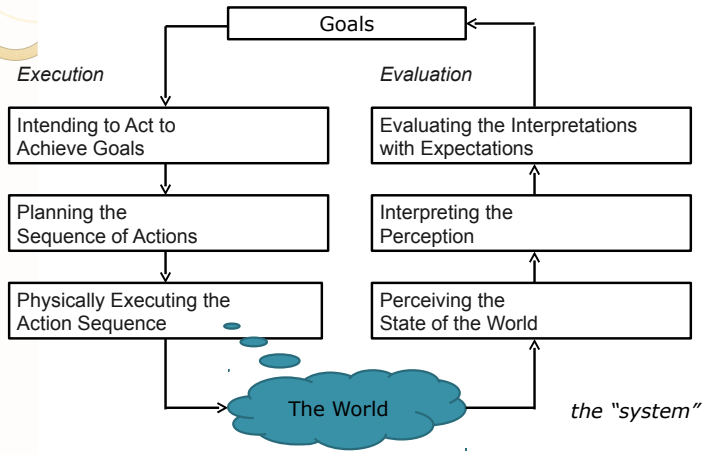
Is it worth the effort?

- suppose there are 1 billion personal computer users
- suppose it takes 5 seconds for a user to recover from 1 error made in 1 day

= 5 billion seconds spent on error recovery

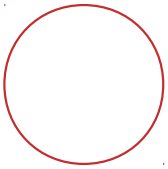
= 160 years

Norman's 7 Stages of Action



Interaction Problems

Adobe Illustrator:
e.g., draw a 2-inch circle with a red boundary



Interaction Difficulties

Gulfs of execution:
effort to turn intentions into actions

Gulfs of evaluation:
effort to interpret feedback

to address these gulfs, need good visibility and feedback

Types of Human Error

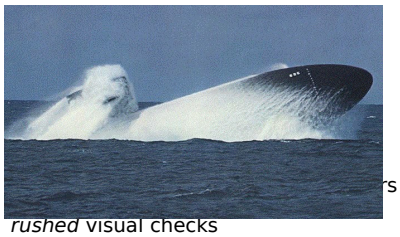
Mistake:
involves *conscious* thought

forming the wrong goal, decision, or judgment

e.g., deciding on the wrong version of file to delete

Example Mistake

USS Greeneville incident:
collided with and sank Japanese fishing vessel

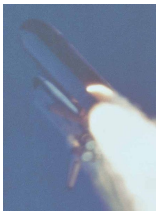


rushed visual checks

disregarded relevant data

Example Mistake

Space Shuttle Challenger incident:
disintegrated at 73 s into launch



social pressure to launch (teacher in space)

Morton Thiokol managers overrode engineers

Rockwell managers did not push their concerns

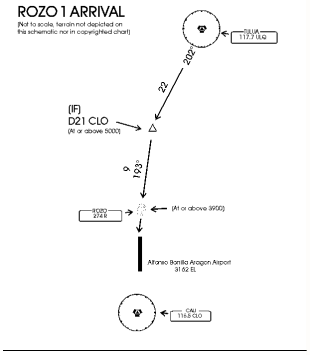
Example Mistake

American Airlines Flight 965:
crashed into a mountain on the approach to Cali, Colombia

erroneously cleared approach waypoints

entered R for waypoint (Rozo intended, but Romeo chosen)

lost situational awareness



Types of Human Error

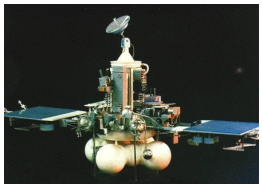
Slip:
involves *everyday* thought

right goal formed, but doing something unintended during the performance

e.g., typing rn not rm to delete a file

Example Slip

Phobos I spacecraft:
- lost contact



- batteries drained
- solar array misoriented
- no navigational lock
- attitude control turned off

- single character omitted in software upload
- no independent double-check

Causes of Slips

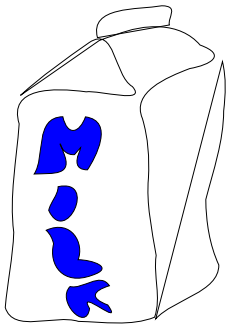
"Capture error":
- when two different sequences of action begin similarly, and the familiar one *captures* the intended one

- e.g., you get in your car on Saturday to go to the store, but end up at work instead

Causes of Slips

"Description error":
- when the intended action has much in common with others that are possible

- e.g., pouring juice on your breakfast cereal, rather than milk



Causes of Slips

"Loss-of-activation error":
- when you forget what to do in the middle of an activity

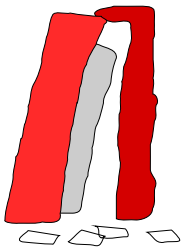
- e.g., you walk from the living room to the bedroom, but forget why you are there in the first place

Causes of Slips

"Perceptual blindness":
when you do not see things that are in plain sight

e.g., you "lost" the stapler, but it is only oriented differently from the norm

more generally, inattentional blindness



Causes of Slips

"Saccadic masking":
when your visual perception is blocked during eye movement

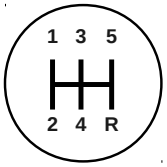
e.g., not noticing the window content already scrolled because it happened during an eye movement

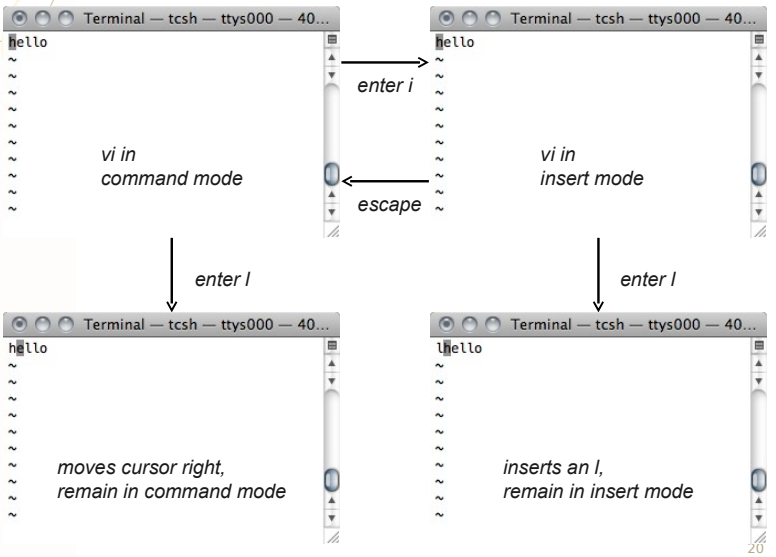
Causes of Slips

"Mode error":
when you think (or forget) something is in one state, but it is actually in another

e.g., car controls

e.g., caps lock key

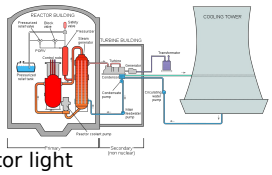




Mode Error

Three Mile Island nuclear plant:

- partial core meltdown
- secondary loop issues
- reactor shutdown
- stuck open relief valve



- misinterpreted relief valve indicator light
- off = powered off, not closed
- not recognizing loss-of-coolant accident

Minimizing Human Errors

Designing in the presence of error:

- normal human behavior is not always direct, accurate, or rational
- understand the causes of error and minimize those causes
- how?

Minimizing Mode Error

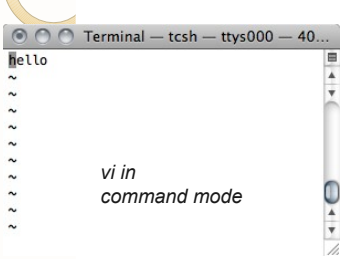
Approaches:

- reduce the number of modes that the user needs to understand
- make modes visible and distinctive

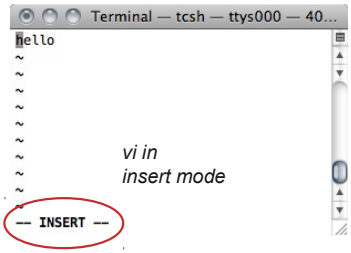
Make Modes Visible

caps lock on

Make Modes Visible



vi in command mode



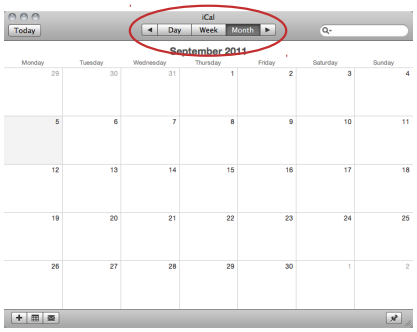
vi in insert mode

Make Modes Visible

each tool is a different mode

how else is the current tool shown?

Make Modes Distinctive



Confirm “Dangerous” Actions

Confirm:
but do not overdo it

user:	remove file “masterpiece”
system:	are you sure?
user:	yes
system:	really?
user:	yes!
system:	file “masterpiece” removed
user:	oops

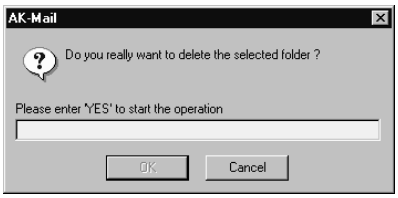
Confirm “Dangerous” Actions

Annoying confirmation:

user:	quit
system:	quit without saving?
user:	yes
system:	are you sure?
user:	yes
system:	return to application?
user:	no!

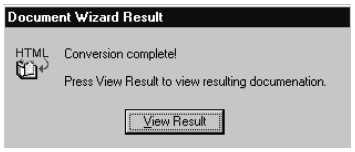
Interface Hall of Shame

Annoying confirmation?



Interface Hall of Shame

Useless post-process confirmation:



Interface Hall of Shame

Don’t forget to give a choice:

Constraining Actions

Constraints:

limitations on actions to prevent problems

but could become annoying in actual usage

e.g., clutch down and turn key to start car

e.g., press brake before shifting out of park

e.g., pressing brake disengages the accelerator

Constraining Actions

Limiting options:

choose only from valid options

- e.g., use combo boxes, sliders, spinners, etc.

gray out options not available in current state

Interface Hall of Shame

Improve Feedback

Detection and correction:

improve feedback to more easily detect errors

give feedback on progress of slow operations

make actions reversible

validate user data

in an application crash,
system suggests corrective actions

Simplify

Reconsider complex mental models:
explicit saving can create complications

determining a destination

user forgetting to save

auto-save feature

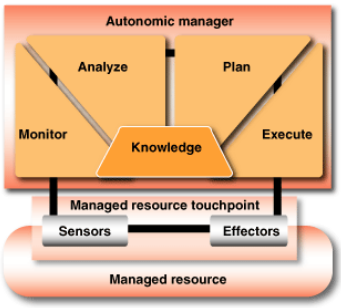
save before quit alert

save replace alert

Reduce Human Actions

Autonomic computing:
system manages itself

self-configuring, -healing, -optimizing, -protecting



© IBM

Alert Messages

clear statement of issue and question

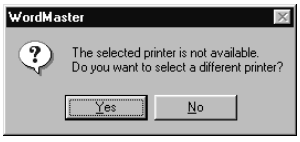
distinctive warning icon

situation and consequences

default choice is safe

Alert Messages

Guideline:
rephrase the message as a question



© Isys Information Architects

Alert Messages

Guidelines:

- tell the truth in plain terms
- don't SHOUT!
- don't be rude
- don't use the word "error"
- don't highlight dangerous buttons in red

Alert Messages

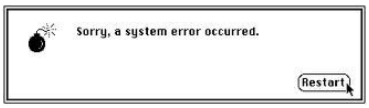
Avoid "violent" language:

- hit
- strike
- punch
- kill
- purge
- execute
- destroy
- boot

Alert Messages

Avoid "violent" metaphors:

- ... of death!
- system bomb, hang, crash, freeze



- viruses, worms, bugs
- hacking, cracking
- ripping, burning

Alert Messages

Avoid cryptic messages:

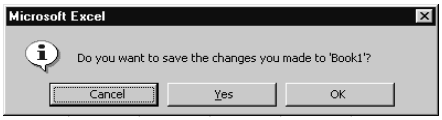
- fatal error
- segmentation fault
- process killed
- core dumped
- kernel panic

Interface Hall of Shame

Interface Hall of Shame

Interface Hall of Shame

Interface Hall of Shame



Interface Hall of Shame

Interface Hall of Shame

Interface Hall of Shame

Reducing Choices

Approach:
reduce too many choices

analysis paralysis

- overanalyzing a situation to the point of not making a decision

Human Efficiency

Hick's Law:
time to make a decision from a set of choices
(if subdivision applies)

average choice reaction time

- $T \approx b \log2(n + 1)$
- n equally probable choices
- constant b determined by experiment

- $T \approx b \sum pi \log2(1/pi + 1) = b \cdot \text{entropy}$
- each choice with probability pi

Human Efficiency

Fitts's Law:
time to point to a target object

- through pointing device or directly

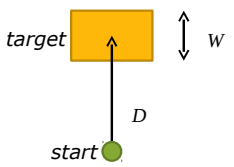
what does the movement time depend on?

- size of target and distance to it

Targeting

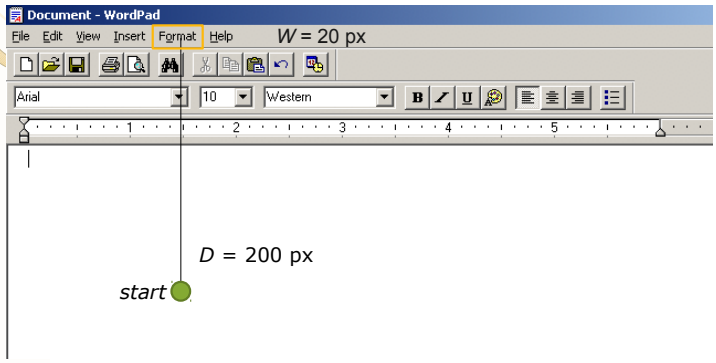
Fitts's Law:
average movement time

- $T \approx a + b \log2(D/W + 1)$
- constants a and b determined by experiment
- distance D from start to center of target
- width W of target along line of motion

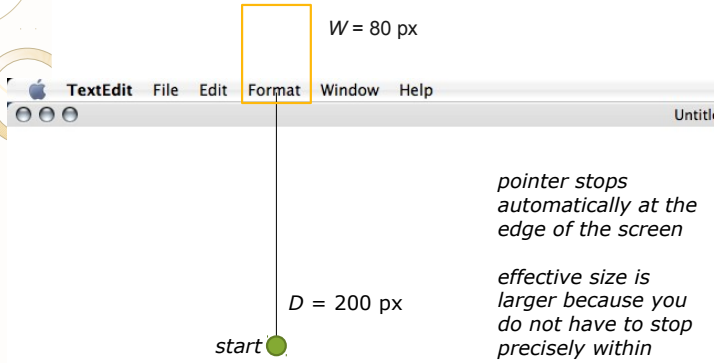


to reduce T, want D low and W high

Menu Targeting Example



top of window menus (Windows)



top of screen menus (Mac)

Menu Targeting Example

Top of window menus:
T = 50 + 150 log2(200/20 + 1) = 569 ms

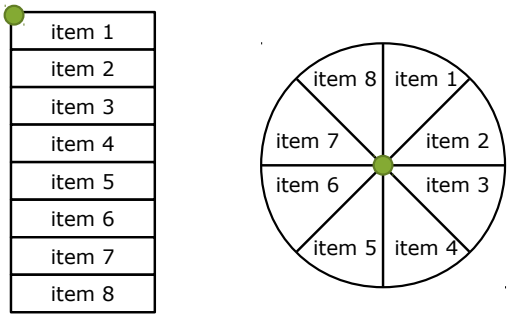
Top of screen menus:
T = 50 + 150 log2(200/80 + 1) = 321 ms

Fitts's Law

Question:
What are the best screen locations to place targets?

Fitts's Law

Question:
Which is typically faster:
linear popup menu or pie (radial) popup menu?



More Information

Books:

The Design of Everyday Things

- D. Norman
- Doubleday, 1988

The Invisible Gorilla

- C. Chabris and D. Simons
- Broadway, 2011

More Information

Books:

Fatal Defect

- I. Peterson
- Vintage, 1995

GUI Bloopers

- J. Johnson
- Morgan Kaufmann, 2000

More Information

Links:

Human Error and the Design of Computer Systems

- http://www.jnd.org/dn.mss/commentary_human_error_and_the_design_of_computer_systems.html

More Information

Links:

The RISKS Digest

- http://catless.ncl.ac.uk/risks

Interface Hall of Shame

- http://homepage.mac.com/bradster/iarchitect/shame.htm

Magic and Software Design

- http://www.asktog.com/papers/magic.html