

Design guidelines and usability heuristics

Human Computer Interaction

Based on slide deck

Part 5: Principles for Design. Design guidelines and usability heuristics

Human Computer Interaction I: Principles and Design

by

Saul Greenberg

Professor

University of Calgary, Canada

*The new slides are marked with a **

Slide deck by Saul Greenberg. Permission is granted to use this for non-commercial purposes as long as general credit to Saul Greenberg is clearly maintained. Warning: some material in this deck is used from other sources without permission. Credit to the original source is given if it is known.

Usability Heuristics

- Avoid common design pitfalls by following 9 design principles
- Inspect an interface for usability problems with these principles

Evaluating Heuristic evaluation

Style guides

Design principles

broad usability statements that guide a developer's design efforts

- use the users language
- provide feedback ...

derived from common design problems across many systems

Heuristic evaluation

Systematic inspection to see if interface complies to guidelines

Method

- 3-5 inspectors
- usability engineers, end users, double experts ...
- inspect interface in isolation (approx. 1-2 hours for simple interfaces)
- compare notes afterwards
 - single evaluator only catches around 35% of usability problems
 - 5 evaluators catch around 75%

Works for paper, prototypes, and working systems

Heuristic evaluation

Advantages

- "minimalist" approach
 - a few guidelines identify many common usability problems
 - easily remembered
 - easily applied with modest effort
- discount usability engineering
 - end users not required
 - cheap and fast way to inspect a system
 - can be done by usability experts, double experts, and end users

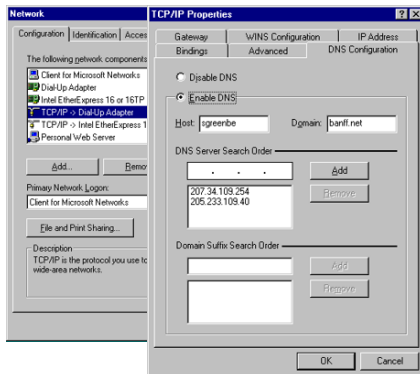
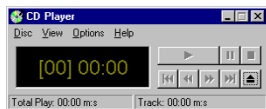
Heuristic evaluation

Problems

- principles are more or less at the motherhood level
 - can't be treated as a simple check-list
 - subtleties involved in their use

1. Simple and natural dialogue

- use the user's conceptual model
- match user's task sequence
- minimize mapping between interface and task semantics

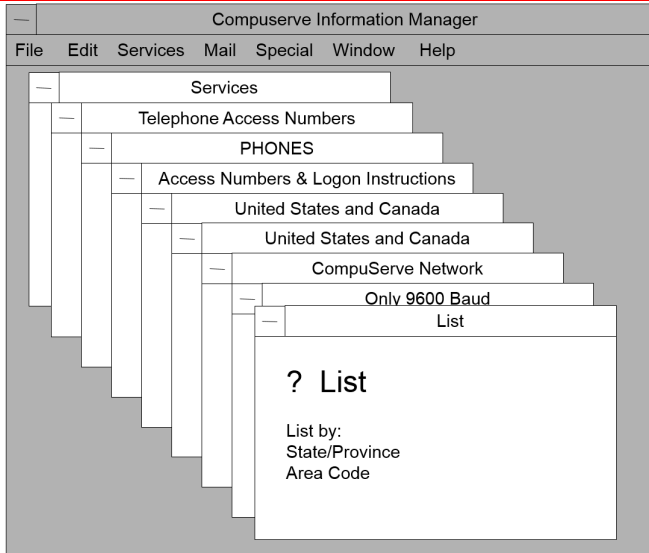


1. Simple and natural dialogue

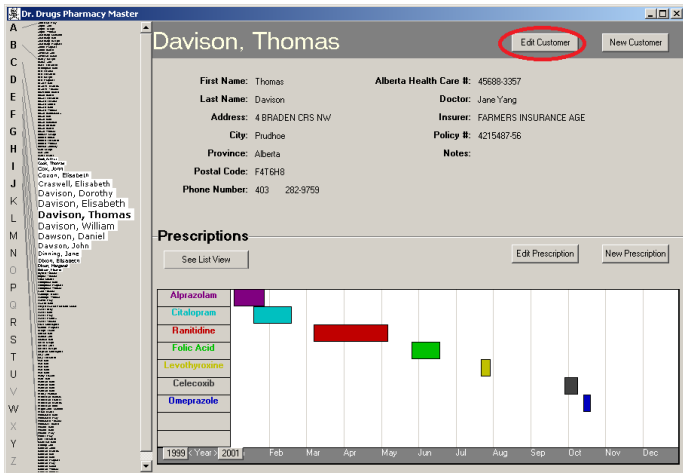
Present exactly the information the user needs

- less is more
 - less to learn, to get wrong, to distract ...
- information should appear in natural order
 - related information is graphically clustered
 - order of accessing information matches user's expectations
- remove or hide irrelevant or rarely needed information
 - competes with important information on screen
- remove modes
- use windows frugally
 - don't add unneeded navigation and windows management

1. Simple and natural dialogue



1. Simple and natural dialogue



Good: Information all in the same place

By previous 481 students Brant LeClercq, Lloyd Yoon, Amy Yang (with permission)

1. Simple and natural dialogue

Dr. Drugs Pharmacy Master

First Name: Thomas
Last Name: Dawson
Address: 4 BRADEN CRS NW
City: Prudhoe
Province: Alberta
Postal Code: F4T6H8
Phone Number: 403
Alberta Health Care ID: 45688-3357
Doctor: Jane Yang
Insurer: FARMERS INSURANCE AG
Policy ID: 4215487-56
Notes:

Prescriptions

See List View Edit Prescription New Prescription

Medication	Start Date	End Date
Alprazolam	Jan 1	Feb 1
Citalopram	Feb 1	Mar 1
Ranitidine	Mar 1	Apr 1
Folic Acid	Apr 1	May 1
Levothyroxine	May 1	Jun 1
Celecoxib	Jun 1	Jul 1
Omeprazole	Jul 1	Aug 1

1999 <Year> 2001 Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec

Good: Information all in the same place

Bad: Special edit mode

1. Simple and natural dialogue

Dr. Drugs Pharmacy Master

Davison, Thomas Edit Customer New Customer

First Name: Thomas **Alberta Health Care #:** 45688-3357
Last Name: Davison **Doctor:** Jane Yang
Address: 4 BRADEN CRS NW **Insurer:** FARMERS INSURANCE AGE
City: Prudhoe **Policy #:** 4215487-56
Province: Alberta **Notes:**
Postal Code: F4T6H8
Phone Number: 403 282-9759

Prescriptions Edit Prescription New Prescription

See List View

Medication	Start Date	End Date
Alprazolam	1999-12-01	2000-01-01
Citalopram	2000-01-01	2000-02-01
Ranitidine	2000-03-01	2000-05-01
Folic Acid	2000-06-01	2000-07-01
Levothyroxine	2000-08-01	2000-08-01
Celecoxib	2000-10-01	2000-10-01
Omeprazole	2000-11-01	2000-11-01

1999 <Year> 2001 Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec

Good: Information all in the same place

By previous 481 students Brant LeClercq, Lloyd Yoon, Amy Yang (with permission)

1. Simple and natural dialogue

Dr. Drugs Pharmacy Master

Davison, Thomas Edit Customer New Customer

First Name: Thomas **Alberta Health Care #:** 45688-3357
Last Name: Davison **Doctor:** Jane Yang
Address: 4 BRADEN CRS NW **Insurer:** FARMERS INSURANCE AGE
City: Prudhoe **Policy #:** 4215487-56
Province: Alberta **Notes:**
Postal Code: F4T6H8
Phone Number: 403 282-9759

Prescriptions Edit Prescription New Prescription

See List View

Drug Name
Manufacturer **Doctor**
Quantity **Units**
Price/Unit **Refill Quantity**

Save Cancel

Good: Stable parts of the window

Bad: Prescriptions separate from graphics

1. Simple and natural dialogue

The screenshot displays the 'Dr. Drugs Pharmacy Master' application window. The main title bar reads 'Dr. Drugs Pharmacy Master'. The patient's name, 'Davison, Thomas', is prominently displayed at the top right, next to a 'New Customer' button. Below the name, there are two red-bordered boxes. The left box contains personal and contact information: First Name: Thomas, Last Name: Davison, Address: 4 BRADEN CRS NW, City: Prudhoe, Province: Alberta, Postal Code: F4T6H8, Phone Number: 403 282-9759, Health Care #: 45688-3357, Doctor: Jane Yang, Insurer: FARMERS INSURANCE AGE, Policy #: 4215487-56, and Notes. The right box contains prescription-related fields: Drug Name, Manufacturer, Quantity, Price/Unit, Doctor, Units, Refill Quantity, and a 'New Prescription' button. A red arrow points from the text 'Click to display info' to the 'New Prescription' button. Below these boxes is a 'Prescriptions' section with a 'See List View' button. A red arrow points from the text 'Double click to edit (mode buttons gone)' to the 'See List View' button. The main area of the 'Prescriptions' section is a Gantt-style chart showing the duration of various medications over time. The medications listed on the left are Alprazolam, Citalopram, Ranitidine, Folic Acid, Levothyroxine, Celecoxib, and Omeprazole. The chart shows colored bars representing the duration of each medication: Alprazolam (purple, Jan-Feb), Citalopram (cyan, Feb-Mar), Ranitidine (red, Mar-May), Folic Acid (green, Jun-Jul), Levothyroxine (yellow, Aug-Sep), Celecoxib (black, Oct-Nov), and Omeprazole (blue, Nov-Dec). At the bottom, there is a year selector showing '1999 < Year | 2001' and a month selector with buttons for Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, and Dec.

Dr. Drugs Pharmacy Master

Davison, Thomas

New Customer

First Name: Thomas
Last Name: Davison
Address: 4 BRADEN CRS NW
City: Prudhoe
Province: Alberta
Postal Code: F4T6H8
Phone Number: 403 282-9759
Health Care #: 45688-3357
Doctor: Jane Yang
Insurer: FARMERS INSURANCE AGE
Policy #: 4215487-56
Notes:

Drug Name
Manufacturer
Quantity
Price/Unit
Doctor
Units
Refill Quantity
New Prescription

Click to display info

Double click to edit (mode buttons gone)

See List View

Prescriptions

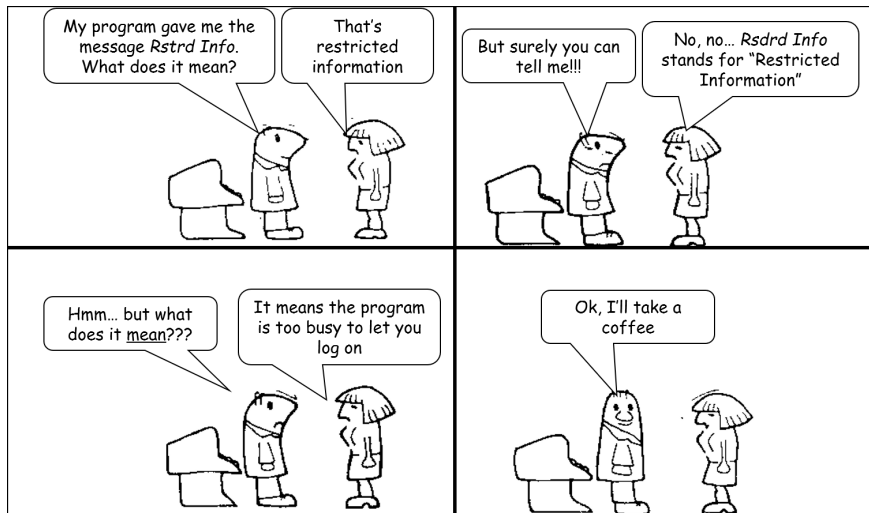
Alprazolam
Citalopram
Ranitidine
Folic Acid
Levothyroxine
Celecoxib
Omeprazole

1999 < Year | 2001

Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec

By previous 481 students Brant LeClerc, Lloyd Yoon, Amy Yang (with permission)

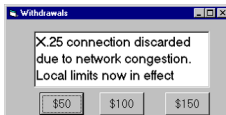
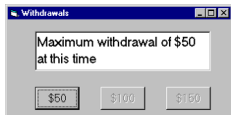
2. Speak the users' language



2. Speak the users' language

Terminology based on users' language for task

- e.g. withdrawing money from a bank machine



Use meaningful mnemonics, icons & abbreviations

- e.g. File / Save
 - Ctrl + S (abbreviation)
 - Alt FS (mnemonic for menu action)



- (tooltip icon)

2. Speak the users' language

Battery Empty Warning

The battery is detected will be run out in several minutes or not present!



If battery is not plugged-in now, you can disable the battery diagnosis in <Battery> page of Configure Notification.

Please click the tray icon and select <Diagnosis Report> mente

Close

Cheap Shop Catalog Store

Donderly software, screen A1.1

Purchaser

Name: Phone:

Postal Code: Province: City:

Delivery Address:

Today's date:

Credit Card No.: for dept use: validation id:

Catalog Item

Number: Quantity: Cost/item: Total:

Balance Owing:

Next Catalog Item (PF5)

Trigger Invoice (PF8)

TurboTax for Windows

User cancelled

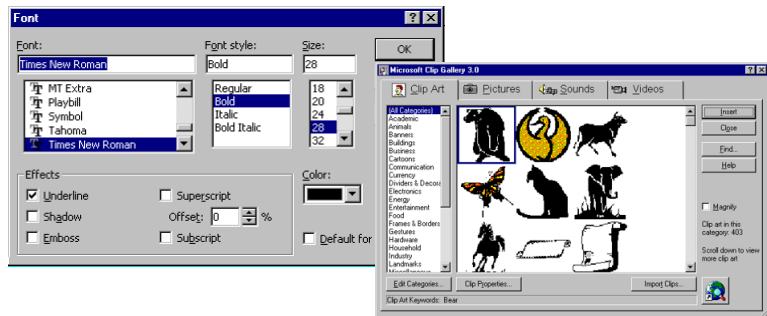
OK

3. Minimize user's memory load

Computers good at remembering, people are not!

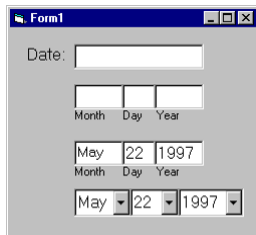
Promote recognition over recall

- menus, icons, choice dialog boxes vs commands, field formats
- relies on visibility of objects to the user (but less is more!)



3. Minimize user's memory load

Give input formats, examples and default values



Form1

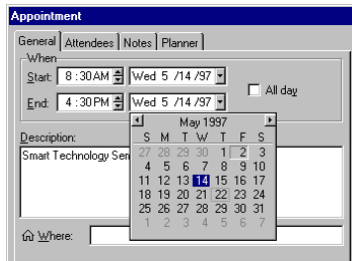
Date:

Month Day Year

May 22 1997

Month Day Year

May 22 1997



Appointment

General Attendees Notes Planner

When

Start: 8:30 AM Wed 5 /14 /97

End: 4:30 PM Wed 5 /14 /97

☐ All day

Description:

Smart Technology Ser

May 1997

S	M	T	W	T	F	S
27	28	29	30	1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31
1	2	3	4	5	6	7

Where:

3. Minimize user's memory load

Small number of rules applied universally

- generic commands
 - same command can be applied to all interface objects
 - interpreted in context of interface object
 - copy, cut, paste, drag 'n drop, ... for characters, words, paragraphs, circles, files
 - context menus

3. Minimize user's memory load

JavaScript



Thank you for your interest in browsing our catalog! It's Easy and it's Efficient! Adobe Acrobat Reader 4.0 uses a 'Pointing Finger' with a 'w' for a mouse pointer whenever you encounter an area where a 'Selection' can be made. When the catalog index page appears, you will notice that the 'Pointing Finger' will appear when you pass over an index item (Product Type) that is selectable. If you click on an item, the pages related to that product will be downloaded to you. Each page has been modularized so that typical download times with a V.90 modem will not exceed 60 seconds with the average download time less than 20 seconds. Depending on your Browser, you may not see a time line, just be patient and the pages will appear. In some cases another index page will appear requiring further selection. The same process should be followed. Using the pager in Acrobat Reader is easy and efficient and in a short time you will be an expert at it. To return to the previous index, simply click your Browser 'Back' button. Two other configurations of mouse pointers are also used by Acrobat Reader. An 'Open Hand' for moving the page around and a 'Magnifier' for zooming in and out while viewing the page. You may select either one from the tool bar at the upper part of the screen. Please carefully jot down the Model Numbers of interest so that they can be entered accurately in the on-line ordering system.

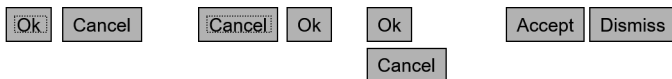
OK

4. Be consistent

Consistent syntax of input

Consistent language and graphics

- same visual appearance across the system (e.g. widgets)
- same information/controls in same location on all windows



Consistent effects

- commands, actions have same effect in equivalent situations
 - predictability

4. Be consistent

These are labels with a raised appearance.

Is it any surprise that people try and click on them?

Subscriber

Name: Tech. Re

Account #: Status:

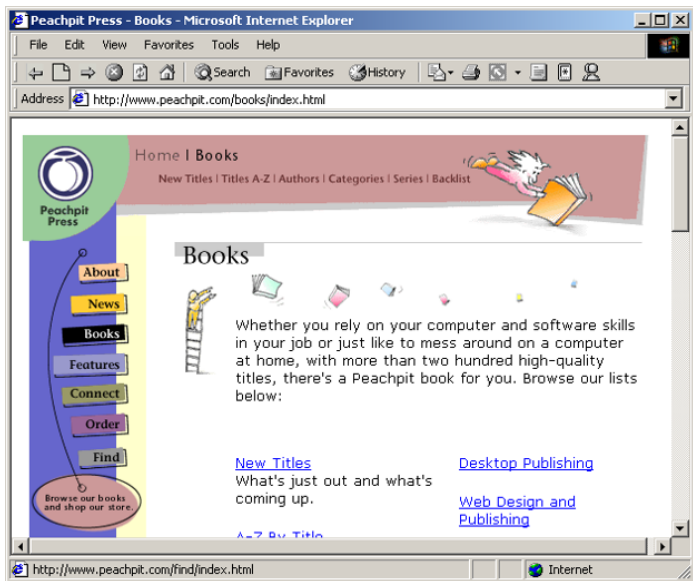
Contact

Telephone: E-Mail:

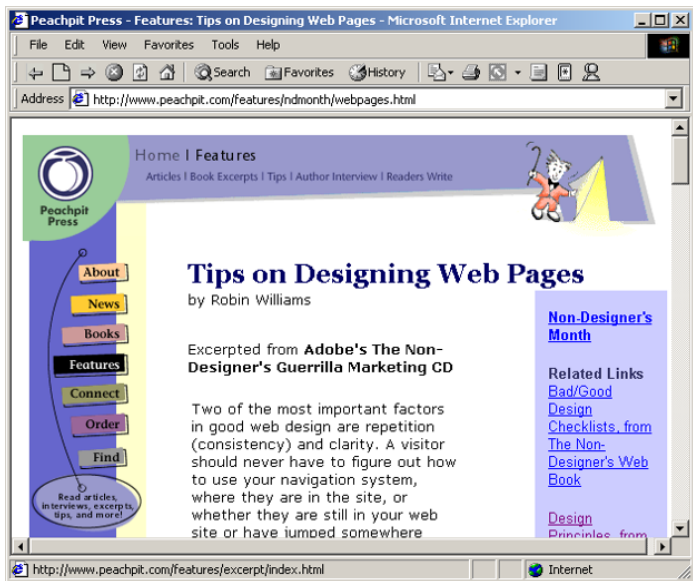
Address: St

Save Cancel

4. Be consistent



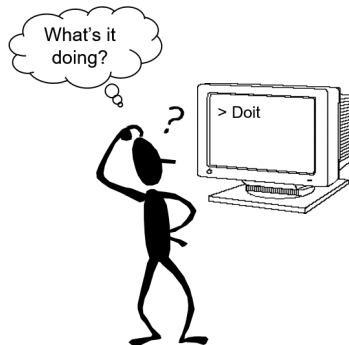
4. Be consistent



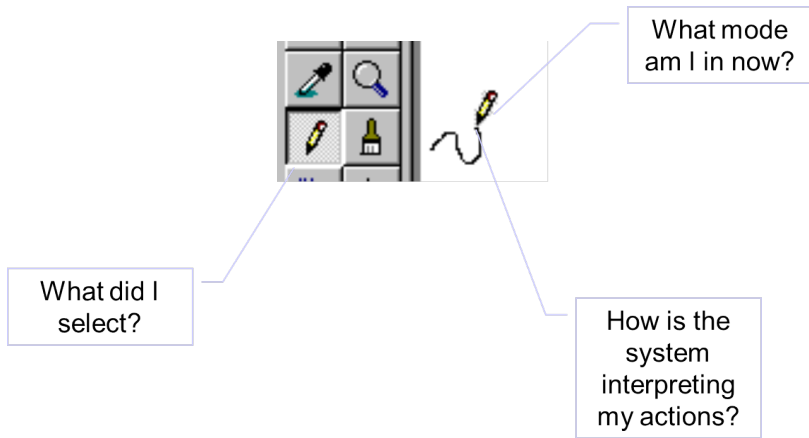
5. Provide feedback

Continuously inform the user about:

- what it is doing
- how it is interpreting the user's input
- user should always be aware of what is going on

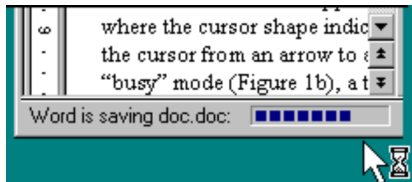
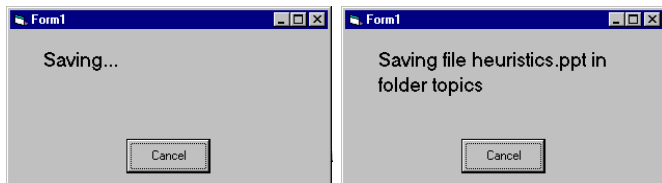


5. Provide feedback



5. Provide feedback

Be as specific as possible, based on user's input



Best within the context of the action

5. Provide feedback

Response time

- how users perceive delays

<0.1s perceived as "instantaneous"

1s user's flow of thought stays uninterrupted, but delay noticed

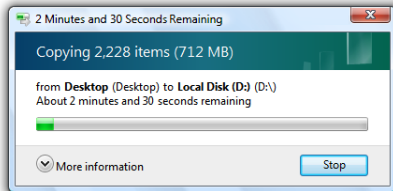
10s limit for keeping user's attention focused on the dialog

>10s user will want to perform other tasks while waiting

5. Provide feedback

Dealing with long delays

- Cursors
 - for short transactions
- Percentage progress bars
 - time left
 - estimated time
- Random
 - for unknown times



Please wait while Windows connects to the "Microsoft" network.



6. Provide clearly marked exits



How do
I get
out of
this?

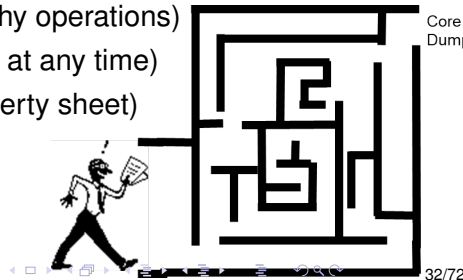
6. Provide clearly marked exits

Users don't like to feel trapped by the computer!

- should offer an easy way out of as many situations as possible

Strategies:

- Cancel button (for dialogues waiting for user input)
- Universal Undo (can get back to previous state)
- Interrupt (especially for lengthy operations)
- Quit (for leaving the program at any time)
- Defaults (for restoring a property sheet)



7. Provide shortcuts

Experienced users - perform frequent operations quickly

Strategies

- keyboard and mouse accelerators
 - abbreviations
 - command completion
 - context menus
 - function keys
 - double clicking vs menu selection
- type-ahead (entering input before the system is ready for it)
- navigation jumps
 - e.g., going to window/location directly, and avoiding intermediate nodes
- history systems
 - WWW: ~ 60% of pages are revisits

7. Provide shortcuts

The image shows a screenshot of the Microsoft PowerPoint 2003 application window. The title bar reads "Microsoft PowerPoint - [M_heuristics]". The menu bar includes File, Edit, View, Insert, Format, Tools, Slide Show, Window, and Help. The toolbar contains various icons for file operations, editing, and presentation controls. The status bar at the bottom indicates "Slide 17 of 45" and the file name "I_involving_user.ppt".

Annotations with red lines pointing to specific UI elements:

- Keyboard accelerators for menus:** Points to the menu bar.
- Customizable toolbars and palettes for frequent actions:** Points to the toolbar.
- Split menu, with recently used fonts on top:** Points to the Font menu.
- Double-click raises toolbar dialog box:** Points to the toolbar.
- Double-click raises object-specific menu:** Points to the context menu (appearing over the slide content).
- Scrolling controls for page-sized increments:** Points to the scroll bar on the right side of the slide.

Slide Content:

7. Provide shortcuts

Experienced users should be able to perform frequently used operations quickly

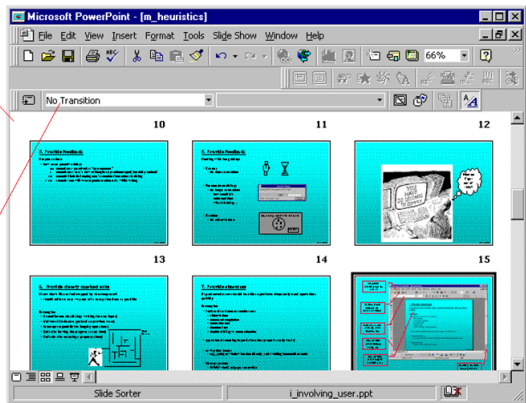
Strategies:

- keyboard and mouse accelerators
 - abbreviations
 - command completion
 - menu shortcuts
 - function keys
 - double clicking vs menu selection
- type-ahead (entering input before the system is ready for it)
- navigation jumps
 - e.g., going to window/location directly, and avoiding intermediate nodes
- history systems
 - WWW: ~60% of pages are revisits

7. Provide shortcuts

Alternate representation for quickly doing different set of tasks

Toolset brought in appropriate to this representation



8. Deal with errors in a positive manner

People will make errors!

Errors we make

- Mistakes
 - conscious deliberations lead to an error instead of correct solution
- Slips
 - unconscious behaviour gets misdirected en route to satisfying goal
 - e.g. drive to store, end up in the office
 - shows up frequently in skilled behaviour
 - usually due to inattention
 - often arises from similar actions

Designing for slips

General rules

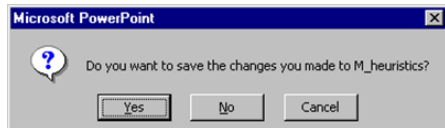
- prevent slips before they occur
- detect and correct slips when they do occur
- user correction through feedback and undo



Types of slips

Capture error

- frequently done activity takes charge instead of one intended
- occurs when common & rarer actions have same initial sequence
 - change clothes for dinner and find oneself in bed (William James, 1890)
 - confirm saving of a file when you don't want to delete it
- minimize by
 - make actions undoable instead of confirmation
 - allows reconsideration of action by use
 - e.g. open trash to undelete a file



Types of slips

Description error

- intended action similar to others that are possible
 - usually occurs when right & wrong objects physically near each other
 - pour juice into bowl instead of glass
 - throw sweaty shirt in toilet instead of laundry basket
 - move file to wrong folder with similar name
- minimize by
 - rich feedback
 - check for reasonable input, etc.
 - undo

Types of slips

Loss of activation

- forget what the goal is while undergoing the sequence of actions
 - start going to room and forget why you are going there
 - navigating menus/dialogs & can't remember what you are looking for
 - but continue action to remember (or go back to beginning)!
- minimize by
 - if system knows goal, make it explicit
 - if not, allow person to see path taken

Types of slips

Mode errors

- people do actions in one mode thinking they are in another
 - refer to file that's in a different directory
 - look for commands / menu options that are not relevant
- minimize by
 - have as few modes as possible (preferably none)
 - make modes highly visible

Generic system responses for errors

General idea: Forcing functions

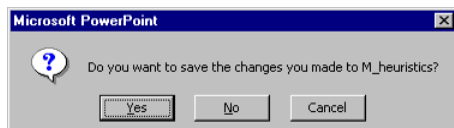
- prevent/mitigate continuation of wrongful action

Gag

- deals with errors by preventing the user from continuing
 - e.g. cannot get past login screen until correct password entered

Warn

- warn people that an unusual situation is occurring
- when overused, becomes an irritant
 - e.g.,
 - audible bell
 - alert box



Generic system responses for errors

Do nothing

- illegal action just doesn't do anything
- user must infer what happened
 - enter letter into a numeric-only field (key clicks ignored)
 - put a file icon on top of another file icon (returns it to original position)

Self-correct

- system guesses legal action and does it instead
- but leads to a problem of trust
 - spelling corrector

Generic system responses for errors

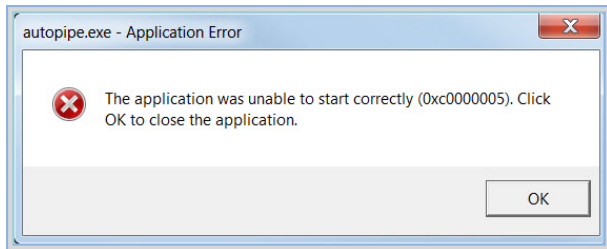
Lets talk about it

- system initiates dialog with user to come up with solution to the problem
 - compile error brings up offending line in source code

Teach me

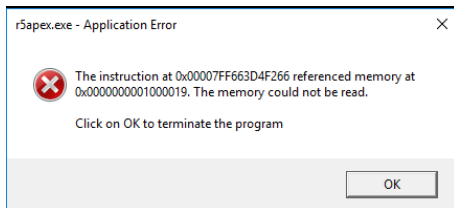
- system asks user what the action was supposed to have meant
- action then becomes a legal one

8. Deal with errors in a positive manner



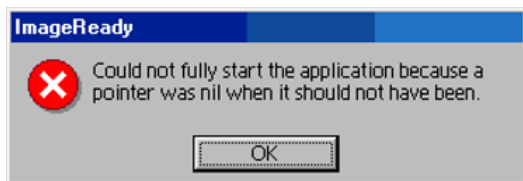
What does "0xc0000005" mean?

8. Deal with errors in a positive manner

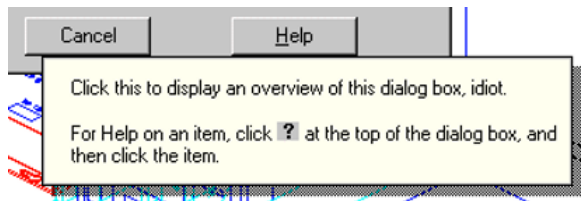


Say what?

8. Deal with errors in a positive manner

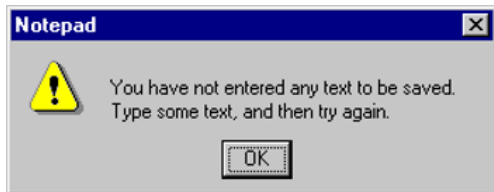


Adobe's *ImageReady*

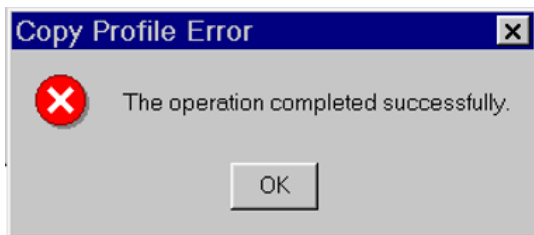


AutoCAD Mechanical

8. Deal with errors in a positive manner



Microsoft Windows' *Notepad*



Microsoft's *NT Operating System*

8. Deal with errors in a positive manner

Provide meaningful error messages

- error messages should be in the user's task language
- don't make people feel stupid

Try again, bonehead!

Error 25

Cannot open this document

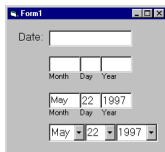
Cannot open "chapter 5" because the application
"Microsoft Word" is not on your system

Cannot open "chapter 5" because the application
"Microsoft Word" is not on your system. Open it with
"Teachtext" instead?

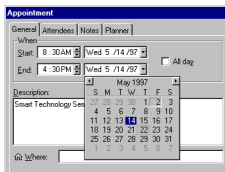
8. Deal with errors in a positive manner

Prevent errors

- try to make errors impossible
- modern widgets: can only enter legal data



A screenshot of a window titled "Form1". It contains a "Date:" label followed by a text input field. Below this, there are three separate input fields for "Month", "Day", and "Year". The "Month" field has a dropdown menu showing "May". The "Day" field has a spinner control showing "22". The "Year" field has a spinner control showing "1997". Below these, there is another set of input fields for "Month", "Day", and "Year", with the "Month" dropdown showing "May", the "Day" spinner showing "22", and the "Year" spinner showing "1997".



A screenshot of a window titled "Appointment". It has tabs for "General", "Attendees", "Notes", and "Planner". The "General" tab is selected. It contains a "When:" section with "Start:" and "End:" labels. The "Start:" field has a dropdown showing "8:30AM" and a date field showing "Wed 5 /14 /97". The "End:" field has a dropdown showing "4:30PM" and a date field showing "Wed 5 /14 /97". There is an "All day" checkbox. Below this, there is a "Description:" section with a text area containing "Smart Technology See". To the right of the text area is a calendar for "May 1997" showing a grid of days. The "Where:" field is at the bottom.

Provide reasonableness checks on input data

- on entering order for office supplies
 - 5000 pencils is an unusually large order. Do you really want to order that many?

9. Provide help

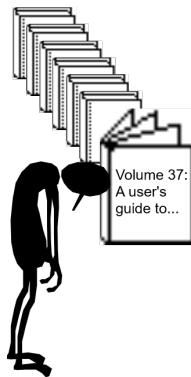
Help is not a replacement for bad design!

Simple systems:

- walk up and use
- minimal instructions

Most other systems

- feature rich
- simple things should be simple
- learning path for advanced features



Documentation and how it is used

Many users do not read manuals

- prefer to spend their time pursuing their task

Usually used when users are in some kind of panic

- paper manuals unavailable in many businesses!
e.g. single copy locked away in system administrator's office
- online documentation better
- good search/lookup tools
- online help specific to current context

Sometimes used for quick reference

- syntax of actions, possibilities ...
- list of shortcuts ...

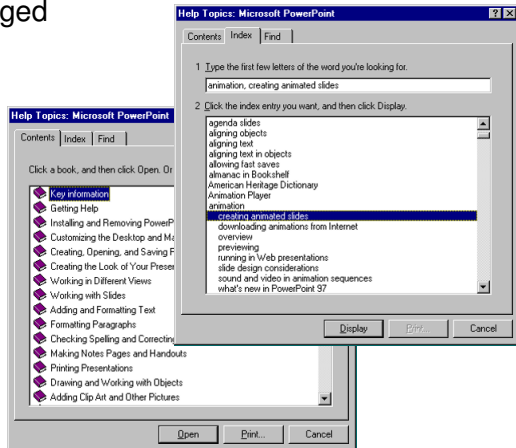
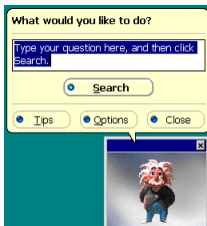
Tutorials and/or getting started manuals

- short guides that people are likely to read when first obtaining their systems
 - encourages exploration and getting to know the system
 - tries to get conceptual material across and essential syntax
- on-line "tours", exercises, and demos
 - demonstrates very basic principles through working examples

Types of help

Reference manuals

- used mostly for detailed lookup by experts
 - rarely introduces concepts
 - thematically arranged
- on-line hypertext
 - search / find
 - table of contents
 - index
 - cross-index



Types of help

Reminders

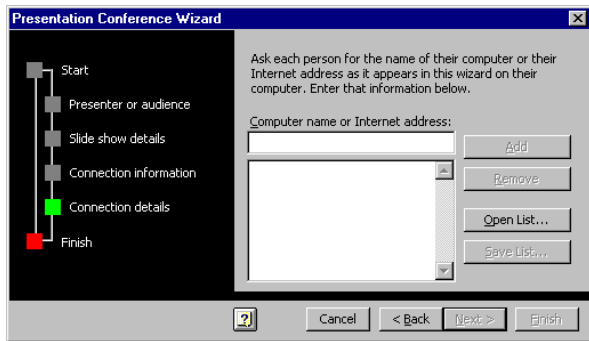
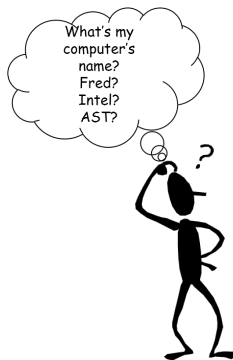
- short reference cards
 - expert user who just wants to check facts
 - novice who wants to get overview of system's capabilities
- keyboard templates
 - shortcuts/syntactic meanings of keys
 - recognition vs. recall
 - capabilities
- tooltips and other context-sensitive help
 - text over graphical items indicates their meaning or purpose



Types of help

Wizards

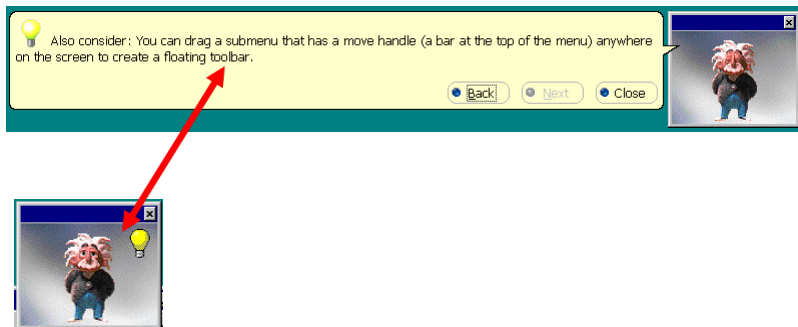
- walks user through typical tasks
- *but* dangerous if user gets stuck



Types of help

Tips

- migration path to learning system features
- also context-specific tips on being more efficient
- must be "smart", otherwise boring and tedious



Evaluating Heuristic evaluation

Problems found by a single inspector

Problems found by multiple inspectors

Individuals vs. teams

Self guided or scenarios?

Problems found by a single inspector

Average over six case studies

- 35% of all usability problems
- 42% of the major problems
- 32% of the minor problems

Not great, but

- finding some problems with one evaluator is *much* better than finding no problems with no evaluators!



Problems found by a single inspector

Varies according to

- difficulty of the interface being evaluated
- the expertise of the inspectors

Average problems found by:

- novice evaluators - 22%
 - no usability expertise
- regular specialists - 41%
 - expertise in usability
- double specialists - 60%
 - experience in both usability and the particular kind of interface being evaluated
 - also find domain-related problems

Tradeoff

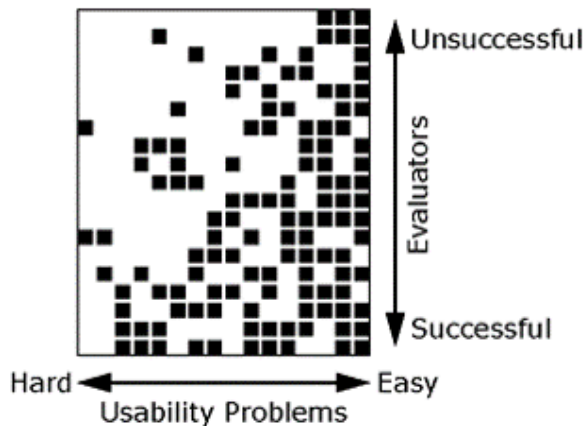
- novices poorer, but cheaper!



Problems found by a single inspector

Evaluators miss both easy and hard problems

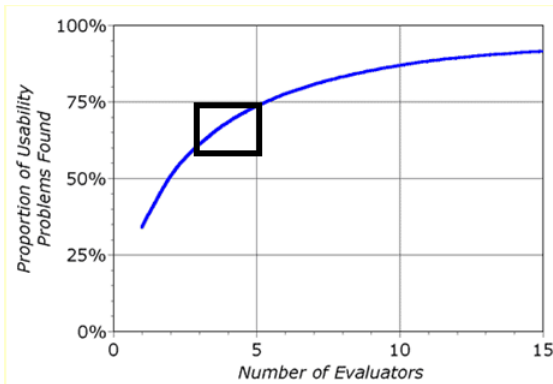
- "best" evaluators can miss easy problems
- "worse" evaluators can discover hard problems



Problems found by multiple inspectors

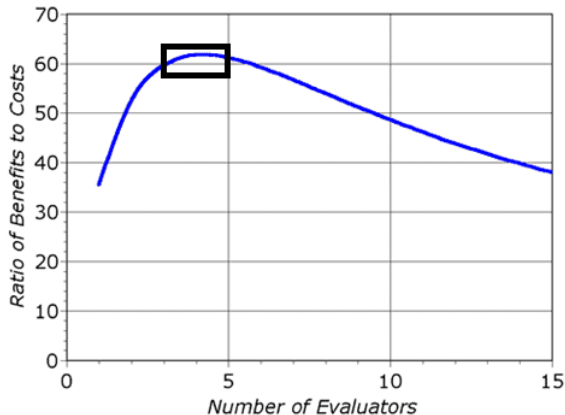
3-5 evaluators find 66-75% of usability problems

- different people find different usability problems
- only modest overlap between the sets of problems found



Problems found by multiple inspectors

Where is the best cost/benefit?



Individuals vs teams

Nielsen

- recommends individual evaluators inspect the interface alone

Why?

- evaluation is not influenced by others
- independent and unbiased
- greater variability in the kinds of errors found
- no overhead required to organize group meetings



Self Guided vs Scenario Exploration

Self-guided

- open-ended exploration
- not necessarily task-directed
- good for exploring diverse aspects of the interface, and to follow potential pitfalls

Scenarios

- step through the interface using representative end user tasks
- ensures problems identified in relevant portions of the interface
- ensures that specific features of interest are evaluated
- but limits the scope of the evaluation - problems can be missed

Other guidelines

Style guides

Guidelines published by producers of graphical user interfaces (GUIs)

- example
 - Microsoft Windows
 - Apple
 - all major software platforms have published guidelines for user interface design

Describes the "look and feel" of the GUI **Good, but hard too follow**

- GUI and widget specific
- vast number of guidelines
- may miss fundamental design principles

Microsoft Windows

- User experience guidelines for Windows-based desktop applications

`https://docs.microsoft.com/en-us/windows/desktop/uxguide/guidelines`

- User Interface Principles

`https://docs.microsoft.com/en-us/windows/desktop/appuistart/-user-interface-principles`

Apple

- Human Interface Guidelines

`https://developer.apple.com/design/human-interface-guidelines/`

Android

- Android Design Guidelines

`https://developer.android.com/design`

You know now

Nine principles of design

- Simple and natural dialogue
- Speak the user's language
- Minimize user's memory load
- Be consistent
- Provide feedback
- Provide clearly marked exits
- Provide shortcuts
- Deal with errors in a positive manner
- Provide help

Heuristic evaluation

- Principles can be used to systematically inspect the interface for usability problems

Style guides

Interface Design and Usability Engineering

Goals:

Articulate:
•who users are
•their key tasks

Brainstorm
designs

Refined
designs

Completed
designs

Methods:

Task centered
system design

Participatory
design

User-centered
design

Evaluate
tasks

Psychology
of everyday
things

User involve-
ment
**Representation
& metaphors**

low fidelity
prototyping
methods

Participatory
interaction

Task
scenario
walk-
through

Throw-
away paper
prototypes

Graphical
screen
design

**Interface
guidelines**

**Style
guides**

high
fidelity
prototyping
methods

Testable
prototypes

Usability
testing

**Heuristic
evaluation**

Field
testing

Products:

User and
task descrip-
tions

Throw-
away paper
prototypes

Testable
prototypes

Alpha/beta
systems or
complete
specification

*Bibliography

- Saul Greenberg, **Principles for Design. Design guidelines and usability heuristics**, University of Calgary, Canada
<http://pages.cpsc.ucalgary.ca/~saul/481/>
- Keith Andrews, **Human Computer Interaction, Chapter 8. Usability Inspection Methods**, TU Graz, Austria
<https://courses.isds.tugraz.at/hci/hci.pdf>