user interface design paradigms

What are Paradigms

- Predominant theoretical frameworks or scientific world views
  - e.g., Aristotelian, Newtonian, Einsteinian (relativistic) paradigms in physics
- Understanding HCI history is largely about understanding a series of paradigm shifts
  - Not all listed here are necessarily "paradigm" shifts, but are at least candidates
  - History will judge which are true shifts

Paradigms of interaction

New computing technologies arrive, creating a new perception of the human—computer relationship.

We can trace some of these shifts in the history of interactive technologies.

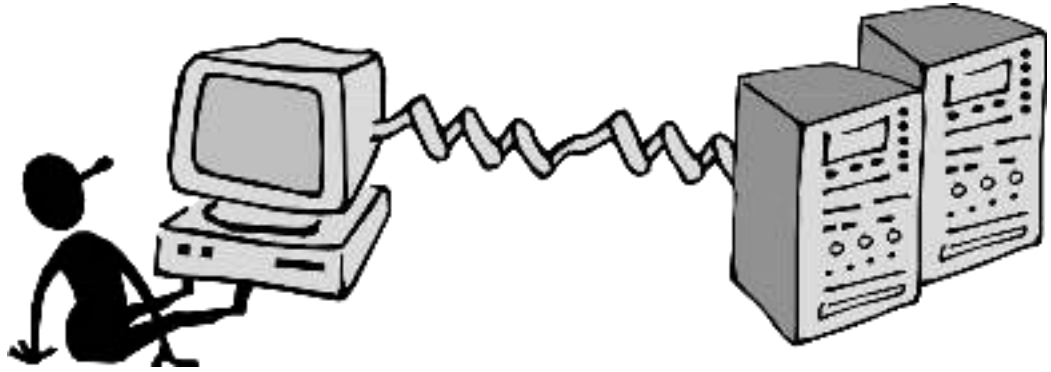
The initial paradigm

- Batch processing

Impersonal computing

Example Paradigm Shifts

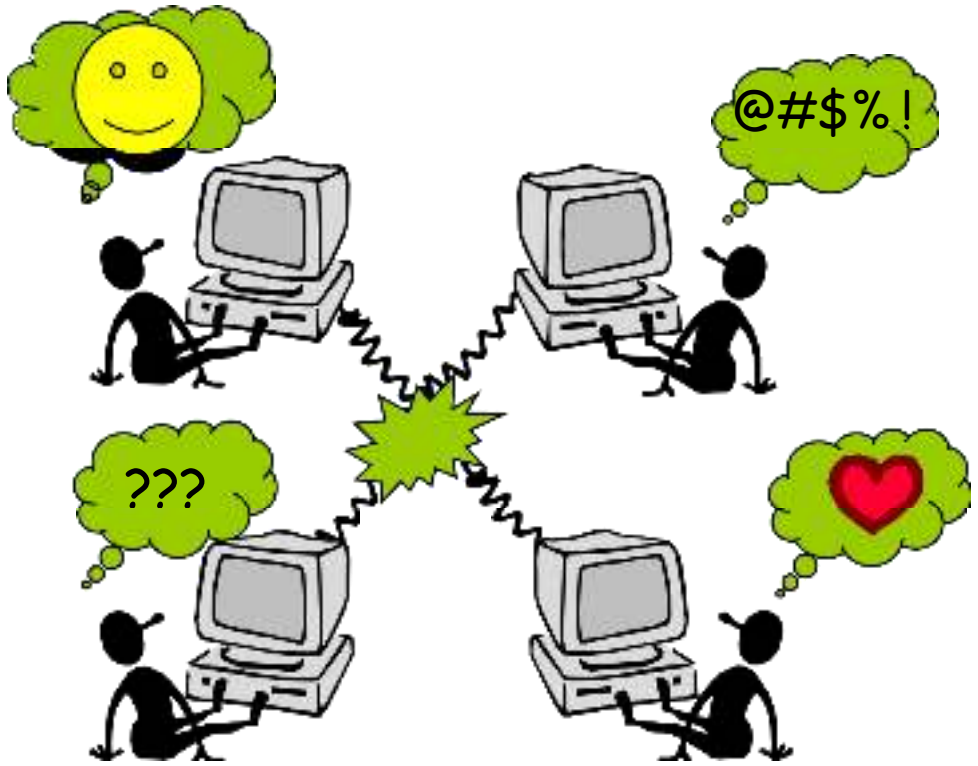
- Batch processing
- Time-sharing



Interactive computing

Example Paradigm Shifts

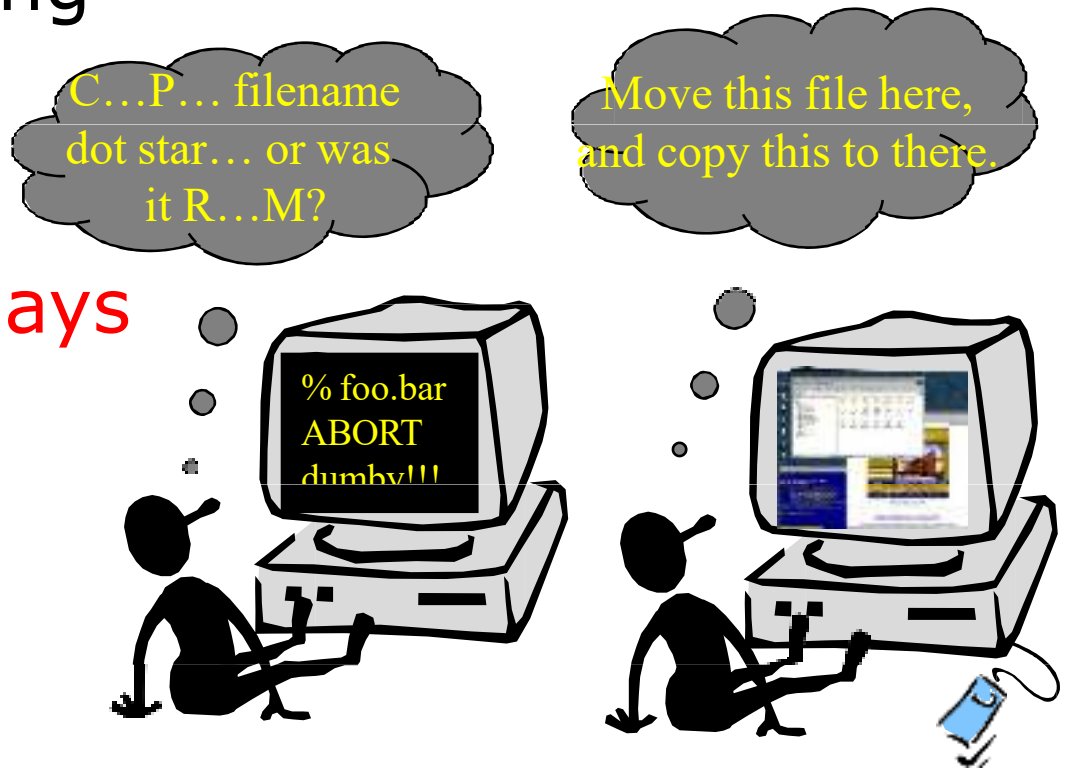
- Batch processing
- Timesharing
- Networking



Community computing

Example Paradigm Shifts

- Batch processing
- Timesharing
- Networking
- Graphical displays



Direct manipulation

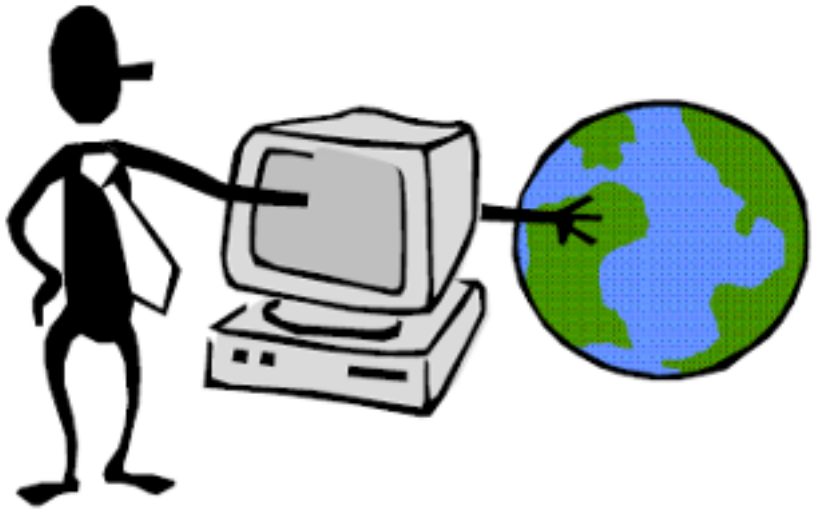
Example Paradigm Shifts

- Batch processing
- Timesharing
- Networking
- Graphical display

Personal computing

Example Paradigm Shifts

- Batch processing
- Timesharing
- Networking
- Graphical display
- WWW

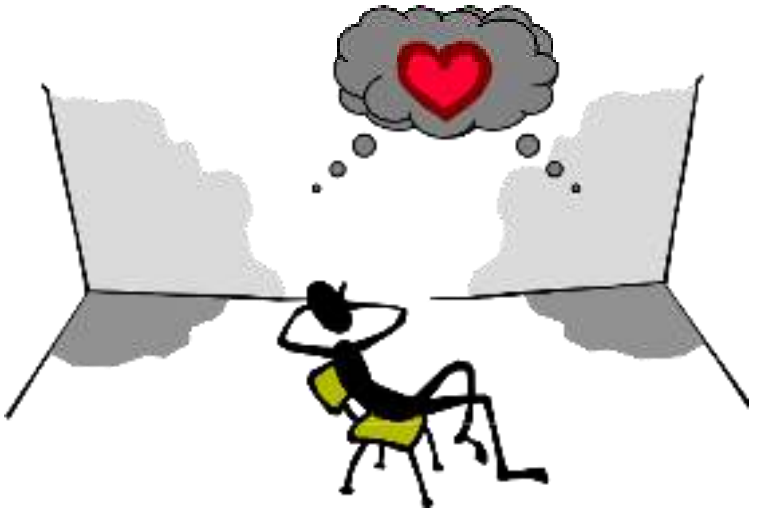


Global information

Example Paradigm Shifts

- Batch processing
- Timesharing
- Networking
- Graphical display
- WWW
- Ubiquitous Computing

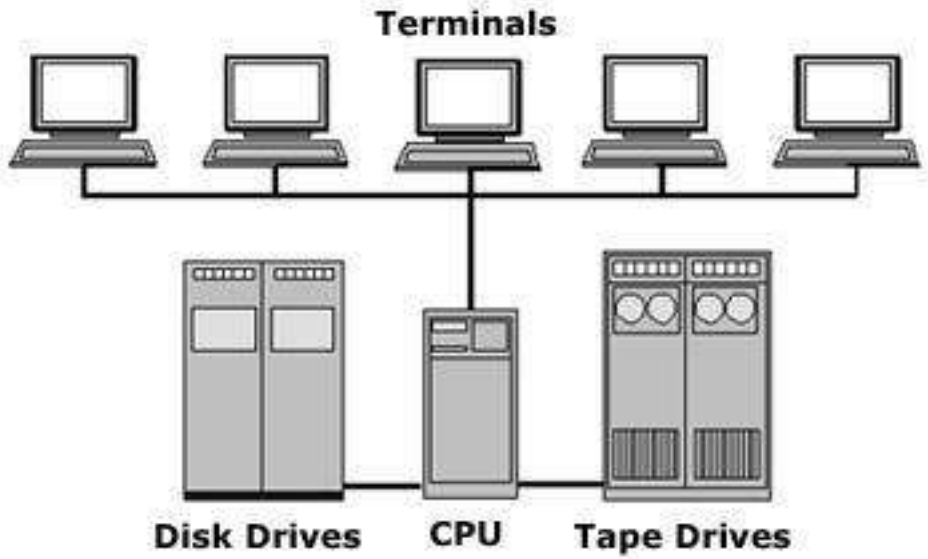
- A symbiosis of physical and electronic worlds in service of everyday activities.



Time-sharing

- 1940s and 1950s – explosive technological growth

- single computer supporting multiple users



Video Display Units

- More suitable medium than paper
- Computers for visualizing and manipulating data

Personal computing

- 1970s – Papert's LOGO language for simple graphics programming by children
- A system is more powerful as it becomes easier to user

Window systems and the WIMP interface

- humans can pursue more than one task at a time
- windows used for dialogue partitioning, to "change the topic"
- 1981 – Xerox Star first commercial windowing system
- windows, icons, menus and pointers now familiar interaction mechanisms

Computer Supported Cooperative Work (CSCW)

- CSCW removes bias of single user / single computer system
- Can no longer neglect the social aspects
- Electronic mail is most prominent success

The World Wide Web

- Hypertext, as originally realized, was a closed system
- Simple, universal protocols (e.g. HTTP) and mark-up languages (e.g. HTML) made publishing and accessing easy
- Critical mass of users lead to a complete transformation of our information economy.

Ubiquitous Computing

"The most profound technologies are those that disappear."

Mark Weiser, 1991

Late 1980's: computer was very apparent

How to make it disappear?

- Shrink and embed/distribute it in the physical world
- Design interactions that don't demand our intention

Design Rules

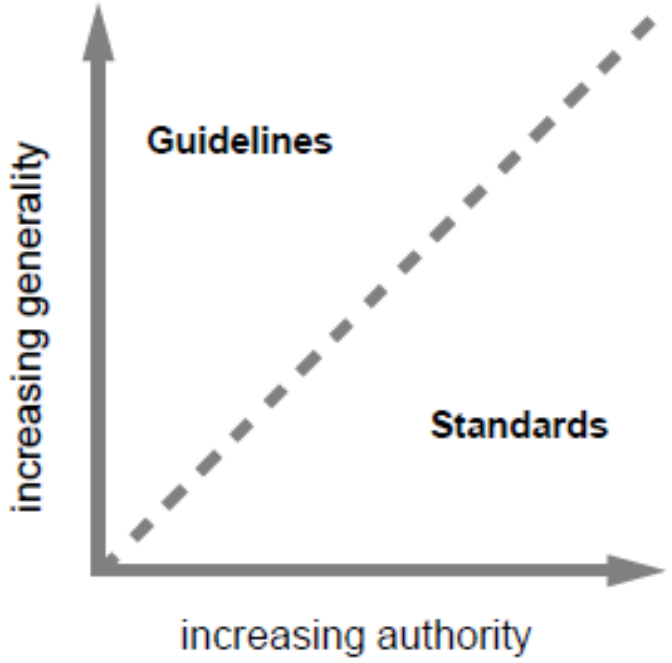
design rules

Designing for maximum usability
– the goal of interaction design

- Principles of usability
  - general understanding
- Standards and guidelines
  - direction for design

types of design rules

- principles
  - abstract design rules
  - low authority
  - high generality
- standards
  - specific design rules
  - high authority
  - limited application
- guidelines
  - lower authority
  - more general application



Principles to support usability

Learnability

the ease with which new users can begin effective interaction and achieve maximal performance

Flexibility

the multiplicity of ways the user and system exchange information

Robustness

the level of support provided the user in determining successful achievement and assessment of goal-directed behaviour

Principles of learnability

Predictability

- determining effect of future actions based on past interaction history
- operation visibility

Principles of learnability (ctd)

Familiarity

- how prior knowledge applies to new system
- guessability; affordance

Generalizability

- extending specific interaction knowledge to new situations

Consistency

- likeness in input/output behaviour arising from similar situations or task objectives

Principles of flexibility

Customizability

- modifiability of the user interface by user (adaptability) or system (adaptivity)

Principles of robustness

Observability

- ability of user to evaluate the internal state of the system from its perceivable representation

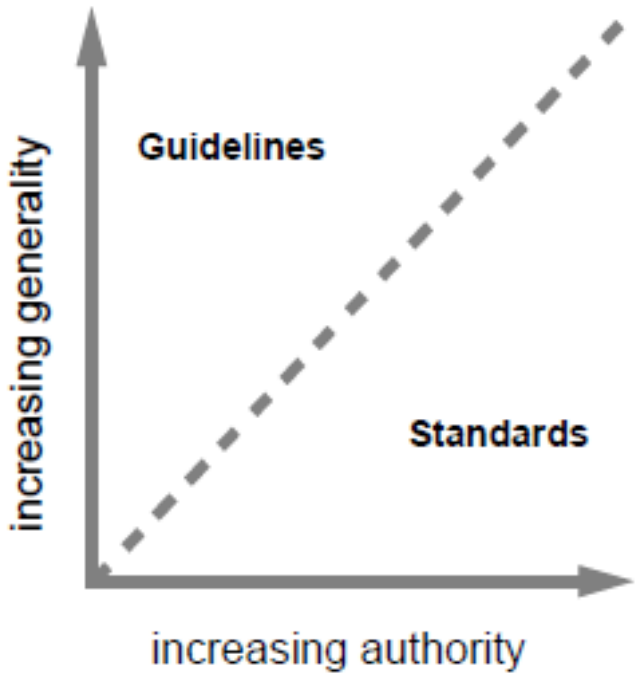
Recoverability

- ability of user to take corrective action once an error has been recognized

Using design rules

Design rules

- suggest how to increase usability
- differ in generality and authority



Standards

- set by national or international bodies to ensure compliance by a large community of designers standards require sound underlying theory and slowly changing technology
- hardware standards more common than software high authority and low level of detail
- ISO 9241 defines usability as effectiveness, efficiency and satisfaction with which users accomplish tasks

Guidelines

- more suggestive and general
- many textbooks and reports full of guidelines
- abstract guidelines (principles) applicable during early life cycle activities
- detailed guidelines (style guides) applicable during later life cycle activities

Golden rules and heuristics

- Useful check list for good design
- Better design using these than using nothing!
- Different collections e.g.
  - Nielsen's 10 Heuristics
  - Shneiderman's 8 Golden Rules
  - Norman's 7 Principles

Shneiderman's 8 Golden Rules

1. *Strive for consistency*
2. *Enable frequent users to use shortcuts*
3. *Offer informative feedback*
4. *Design dialogs to yield closure*
5. *Offer error prevention and simple error handling*
6. *Permit easy reversal of actions*
7. *Support internal locus of control*
8. *Reduce short-term memory load*

Norman's 7 Principles

1. *Use both knowledge in the world and knowledge in the head.*
2. *Simplify the structure of tasks.*
3. *Make things visible.*

4. *Get the mappings right.*
5. *Exploit the power of constraints, both natural and artificial.*
6. *Design for error.*
7. *When all else fails, standardize.*

Heuristics (by Nielsen)

- use simple and natural dialogue sequences
- speak the users' language
- minimize user memory load
- be consistent
- provide feedback
- provide clearly marked exits
- provide shortcuts
- provide good error messages
- prevent errors

Windows Interface Guidelines

•Set of general principles for interface design in Microsoft's software development documentation

- directness
- user in control
- consistency
- forgiveness
- feedback
- aesthetics
- simplicity

Many common elements…

Nielsen

- use simple and natural dialogue sequences
- speak the users language
- minimize user memory load
- be consistent
- provide feedback
- provide clearly marked exits
- provide shortcuts
- provide good error messages
- prevent errors

Shneiderman

- strive for consistency
- enable frequent users to use shortcuts
- offer informative feedback
- design dialogues to yield closure
- offer simple error handling
- permit easy reversal of actions
- reduce short term memory load

Microsoft

- directness
- user in control
- consistency
- forgiveness
- feedback
- aesthetics
- simplicity

Be consistent

Consistency……

- important to enable user to build a reliable model of how the interface works
- makes the interface familiar and predictable by providing a sense of stability
- allows users to transfer existing knowledge to new tasks and focus more on tasks because they need not spend time trying to remember the differences in interaction.
- important through all aspects of the interface, names of commands, layout of information, and operational behaviour.

Many common elements…

Nielsen

- use simple and natural dialogue sequences
- speak the users language
- minimize user memory load
- be consistent
- provide feedback
- provide clearly marked exits
- provide shortcuts
- provide good error messages
- prevent errors

Shneiderman

- enable frequent users to use shortcuts
- offer informative feedback
- design dialogues to yield closure
- offer simple error handling
- permit easy reversal of actions
- support internal locus of control
- Reduce short-term memory load

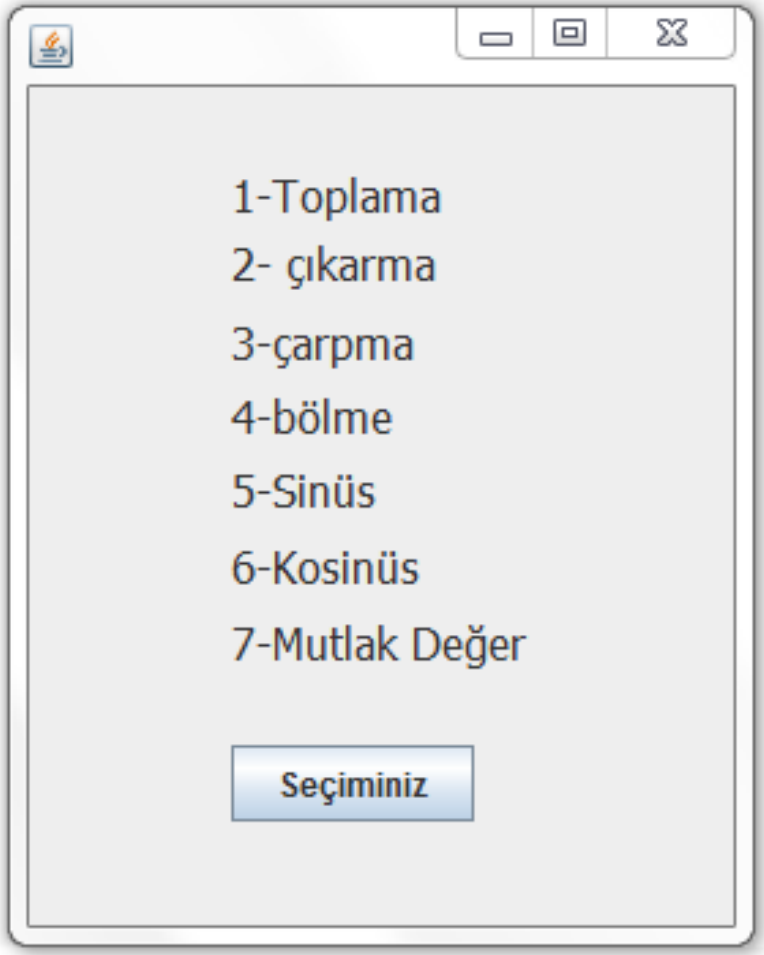
Microsoft

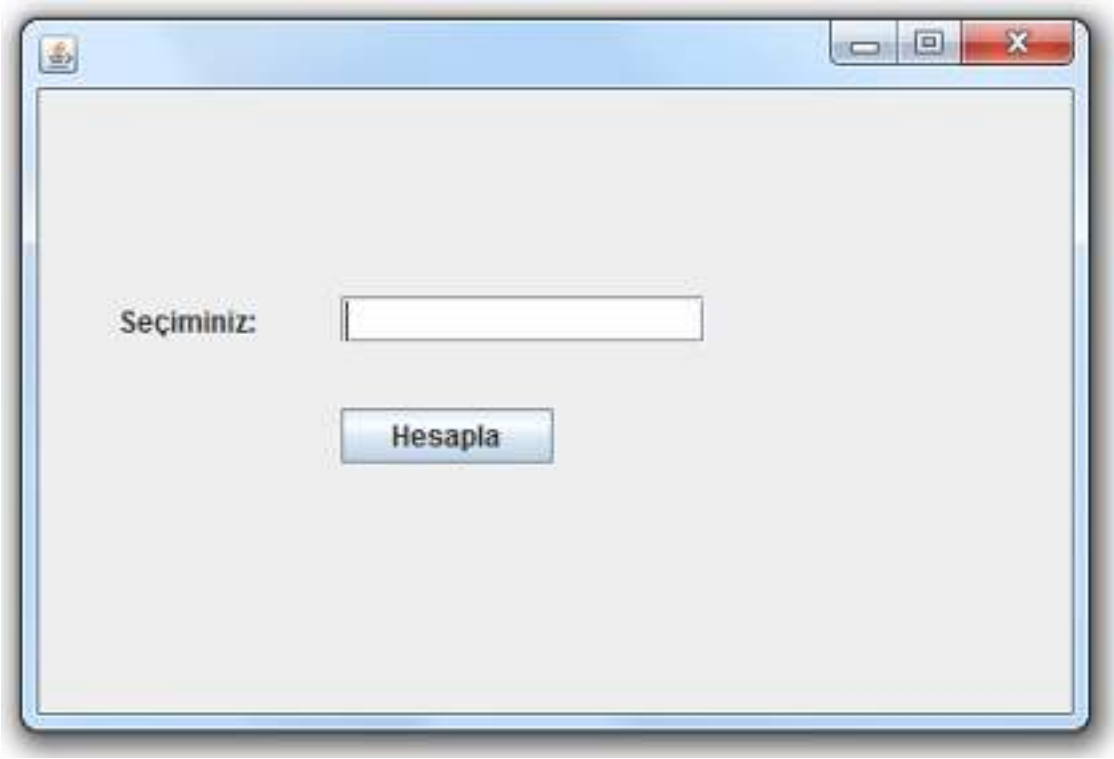
- directness
- user in control
- consistency
- forgiveness
- feedback
- aesthetics
- simplicity

Reduce memory load

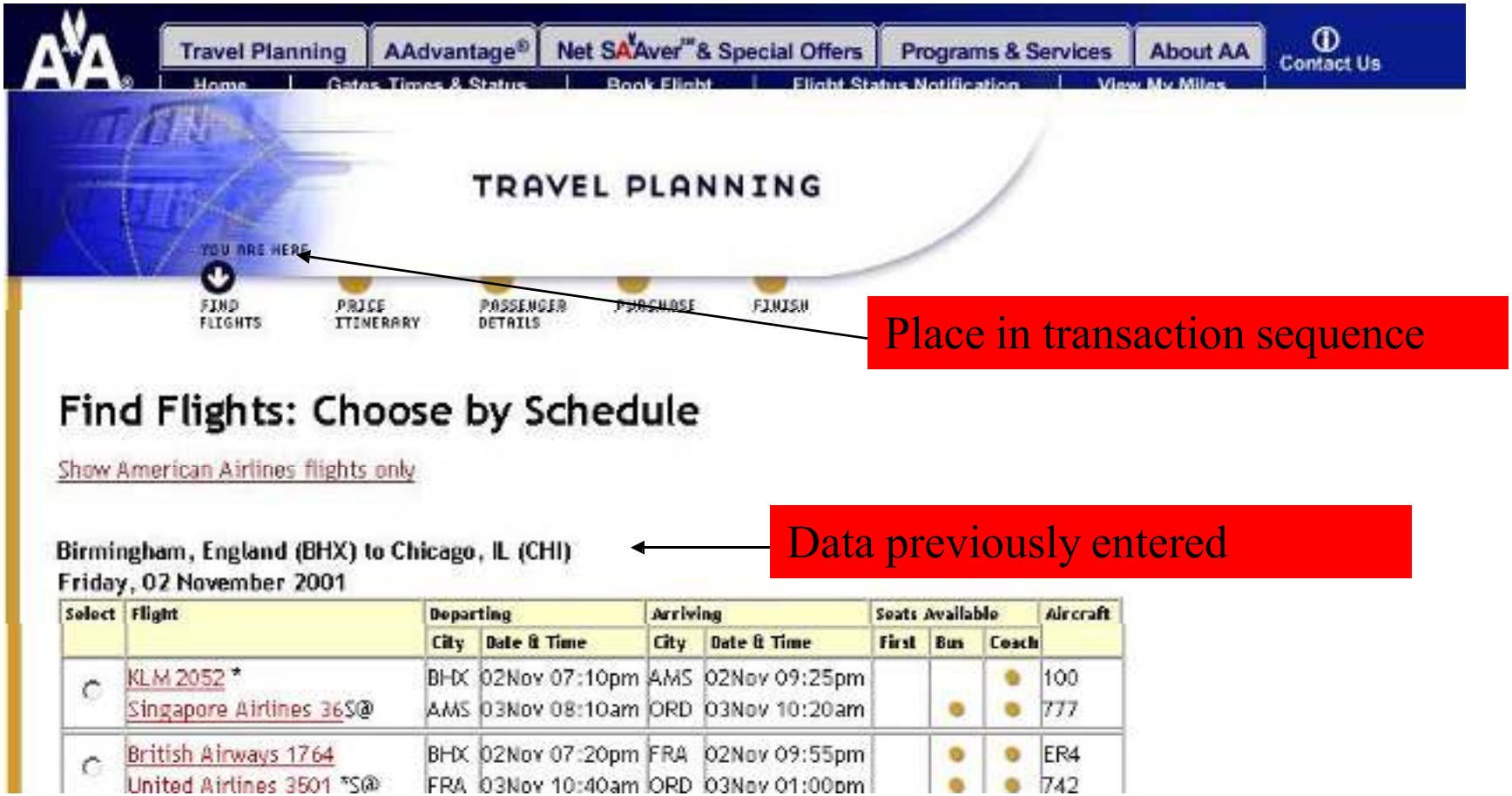
Minimize user memory load

- **Basic rule**: don't expect the user to remember what has already been done. Make this visible at the interface
- If a command is made up of a number of pieces of data entered by the user in sequence, display these rather than expecting the user to remember the data already entered
- Help the user remember where they are in a transaction sequence – Menu 2/5 Step 1 - 4


1-Toplama
2- çıkarma
3-çarpma
4-bölme
5-Sinüs
6-Kosinüs
7-Mutlak Değer
Seçiminiz


Seçiminiz:
Hesapla

Example: American Airlines site



Many common elements…

Nielsen

- use simple and natural dialogue sequences
- speak the users language
- minimize user memory load
- be consistent
- provide feedback
- provide clearly marked exits
- provide shortcuts
- provide good error messages
- prevent errors

Shneiderman

- strive for consistency
- enable frequent users to use shortcuts
- offer informative feedback
- design dialogues to yield closure
- offer simple error handling
- permit easy reversal of actions
- support internal locus of control
- reduce short term memory load

Microsoft

- directness
- user in control
- consistency
- forgiveness
- feedback
- aesthetics
- simplicity

Feedback to user

Feedback from the system

- Every action the user makes should produce a perceptible response.
- The intention is to reduce user uncertainty that the system has:
  - received the last input,
  - is currently doing something about it,
  - or is waiting for the next input.
- Commands should result in some visible change to the interface
  - E.g ‘mail has been sent’ in response to a ‘Send’ command

Feedback: Response Time

- Provide ‘system busy’ feedback if time will exceed a few seconds or is unpredictable
- Provide indication of how many transactions remain, for example as a bar chart or as a percentage.

Many common elements…

Nielsen

- use simple and natural dialogue sequences
- speak the users language
- minimise user memory load
- be consistent
- provide feedback
- provide clearly marked exits
- provide shortcuts
- provide good error messages
- prevent errors

Shneiderman

- strive for consistency
- enable frequent users to use shortcuts
- offer informative feedback
- design dialogues to yield closure
- Offer error prevention and simple error handling
- permit easy reversal of actions
- support internal locus of control
- reduce short term memory

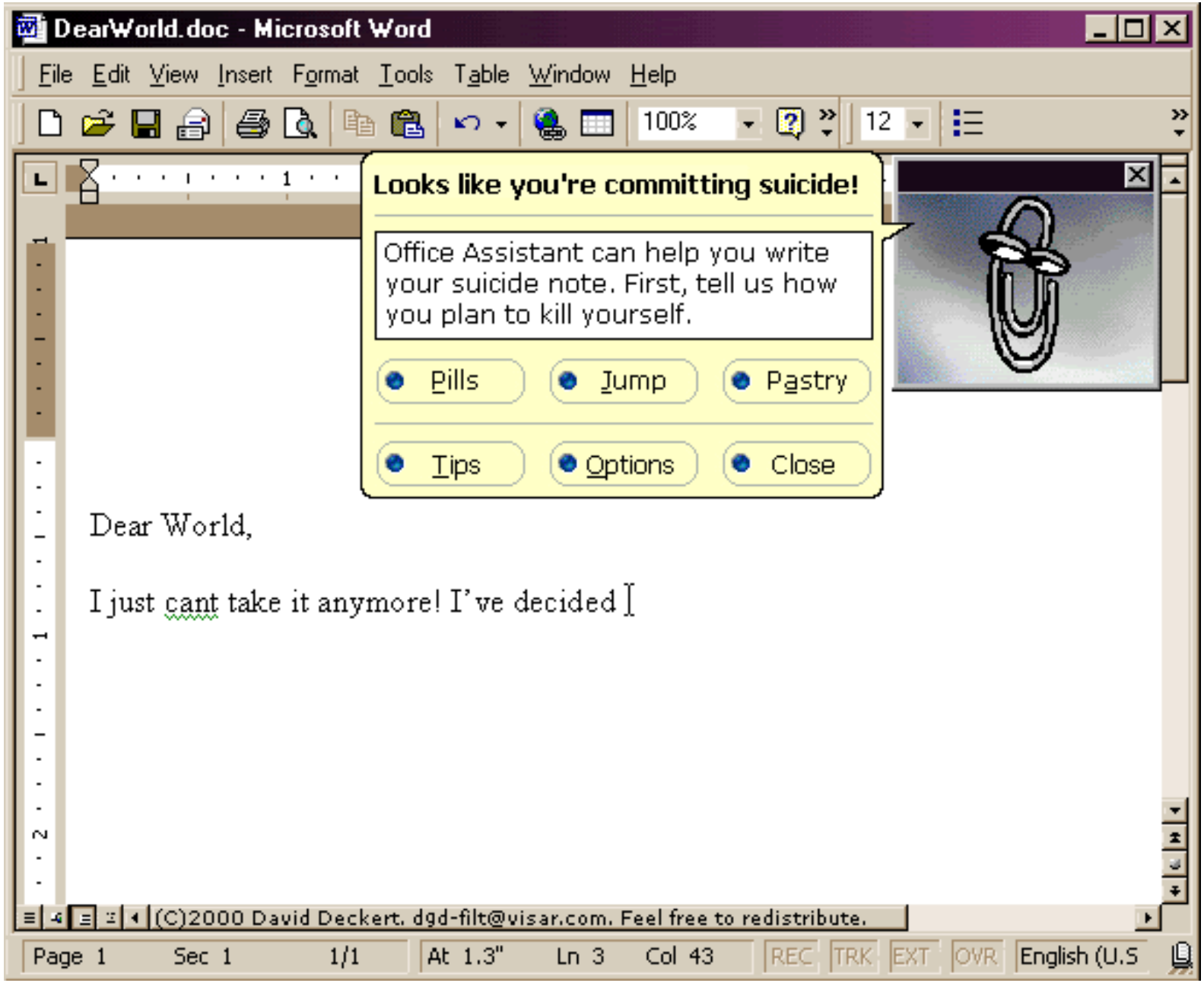
Microsoft

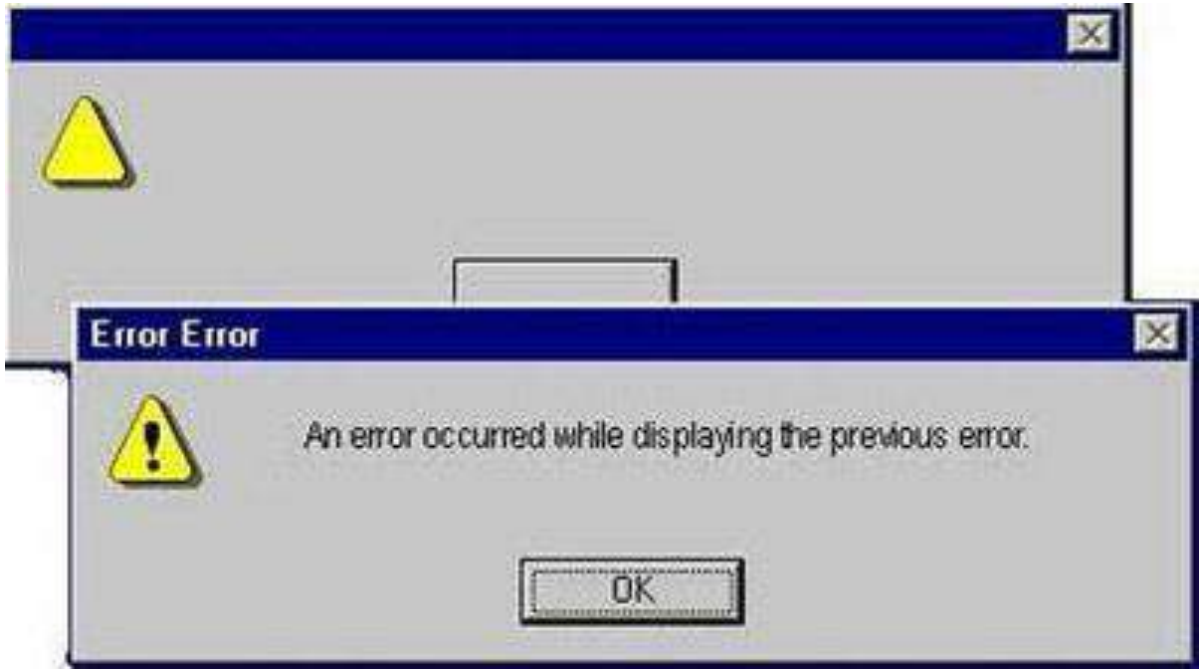
- directness
- user in control
- consistency
- forgiveness
- feedback
- aesthetics
- simplicity

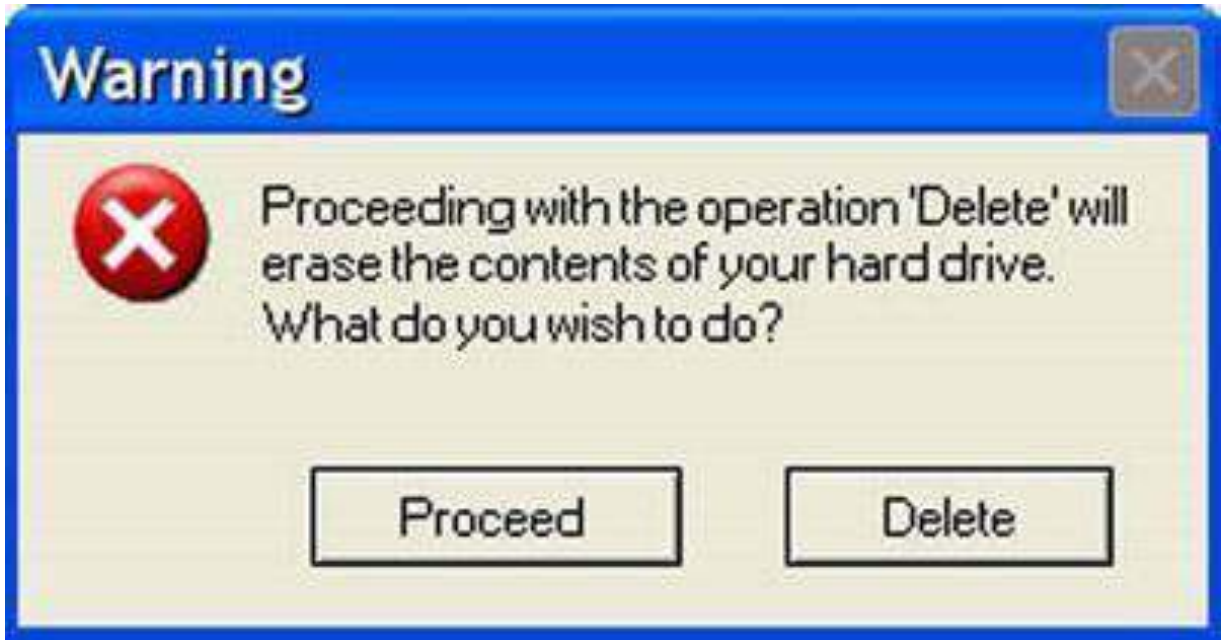
Appropriate user support

Appropriate user support

- HELP messages
  - important to recognise different types of help;
  - should be available when required and context-specific;
  - can the user get help about what responses are possible at a given point in a dialogue.
- ERROR messages
  - should explain what is wrong and what corrective action is required;
  - should use ‘jargon’ familiar to the user;
  - often this support is poorly designed in terms of what information is given to the user.


DearWorld.doc - Microsoft Word
File Edit View Insert Format Tools Table Window Help
100%
12
Looks like you're committing suicide!
Office Assistant can help you write your suicide note. First, tell us how you plan to kill yourself.
Pills
Jump
Pastry
Tips
Options
Close
Dear World,
I just cant take it anymore! I've decided
(C)2000 David Deckert. dgd-filt@visar.com. Feel free to redistribute.
Page 1 Sec 1 1/1 At 1.3" Ln 3 Col 43 REC TRK EXT OVR English (U.S


Error Error
An error occurred while displaying the previous error.
OK


Warning
Proceeding with the operation 'Delete' will erase the contents of your hard drive. What do you wish to do?
Proceed
Delete

Many common elements…

Nielsen

- use simple and natural dialogue sequences
- speak the users language
- minimise user memory load
- be consistent
- provide feedback
- provide clearly marked exits
- provide shortcuts
- provide good error messages

Shneiderman

- strive for consistency
- enable frequent users to use shortcuts
- offer informative feedback
- design dialogues to yield closure
- offer simple error handling
- permit easy reversal of actions
- support internal locus of control
- reduce short term memory load

Microsoft

- directness
- user in control
- consistency
- forgiveness
- feedback
- aesthetics
- simplicity

Flexibility

Flexibility

- Provide alternative means of achieving the same goal which match different models of how the interface works.
  - e.g. word selection: cursor to start of word and double click, click and drag, click and shift-click.
  - e.g. word deletion: word highlighted and Control +X key, select ‘Cut’ menu option, backspace.

Minimal user input

- Balance between number of keystrokes or mouse movements/clicks and memory load.
- Reducing keying errors increases speed of data entry.
- Allow selection from a list rather than typing in a value (recognise rather than recall).
- Edit a command that has produced an error rather than retyping the command.
- Do not request input of information which can be derived automatically or which has been entered previously.
- Use default values.

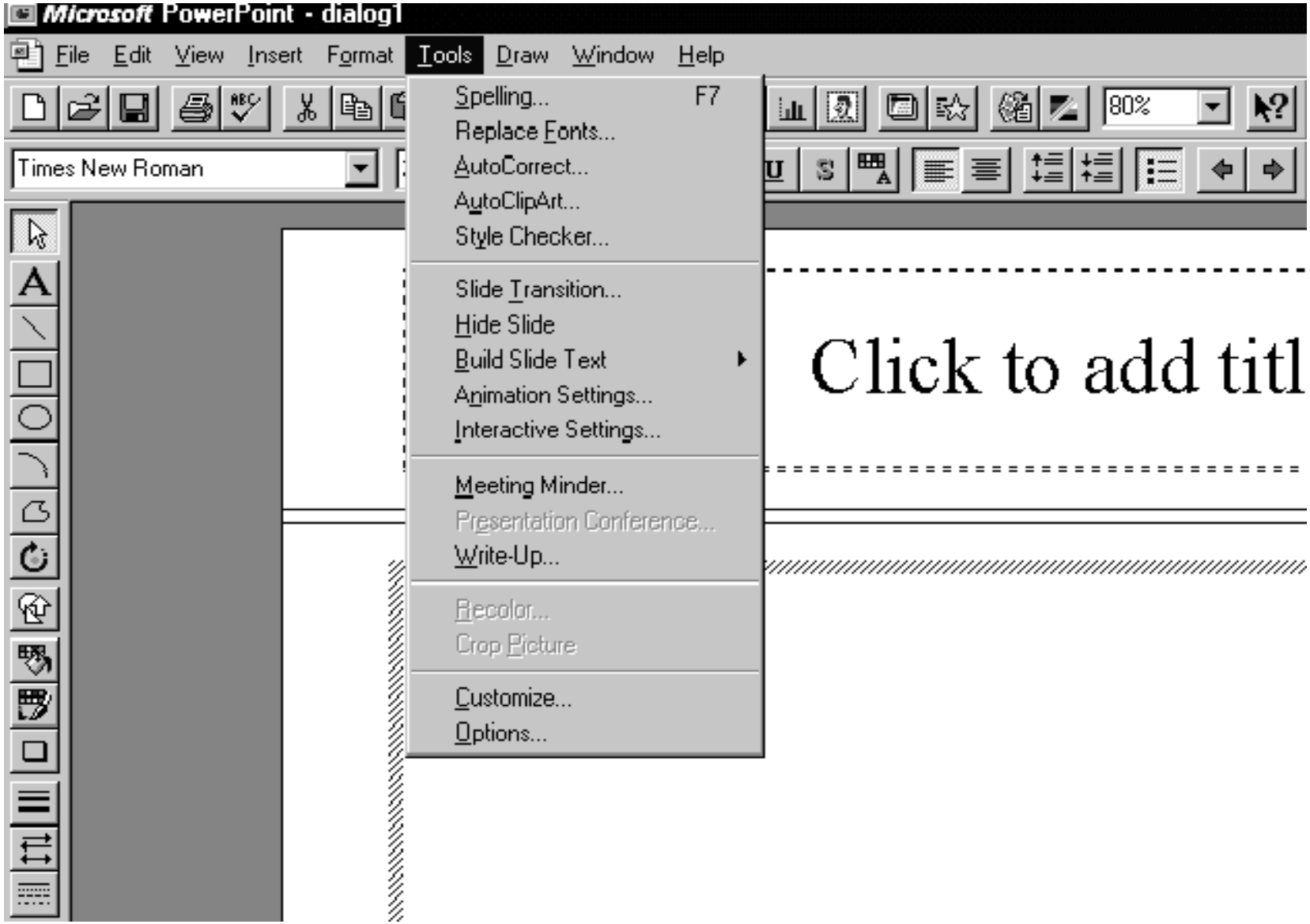
Menus

- Usually a collection of actions, structured into a list from which a user chooses
- Actions applied to objects
  - Explicitly selected by user – format + font… [selected text]
  - Pop-up menu over selected object shows common actions on that object

- Actions may be represented
  - by text (e.g pull-down menu)
  - by icons (e.g toolbar)

Overloading menus

- Most common Windows applications use an 'anything, anytime' approach – i.e., all commands are available to the user at all times
- Leads to large, cumbersome menu structures where the user can forget how to find a particular command
- Toolbars attempt to provide shortcuts to frequently used items
  - order of icons in toolbars different from items in pull-down menus representing same actions
- Many CAD systems use an alternative, "mode" approach where a general type of operation, or task is selected
  - Only a restricted set of menus relevant to that operation are displayed
- This approach is now used in some MS applications


Microsoft PowerPoint - dialog1
File Edit View Insert Format Tools Draw Window Help
Spelling... F7
Replace Fonts...
AutoCorrect...
AutoClipArt...
Style Checker...
Slide Transition...
Hide Slide
Build Slide Text
Animation Settings...
Interactive Settings...
Meeting Minder...
Presentation Conference...
Write-Up...
Recolor...
Crop Picture
Customize...
Options...
80%
Times New Roman
Click to add titl

Menu Structure

'Structures should reflect users expectations.. and support users flow of work' (ISO 9241/14)

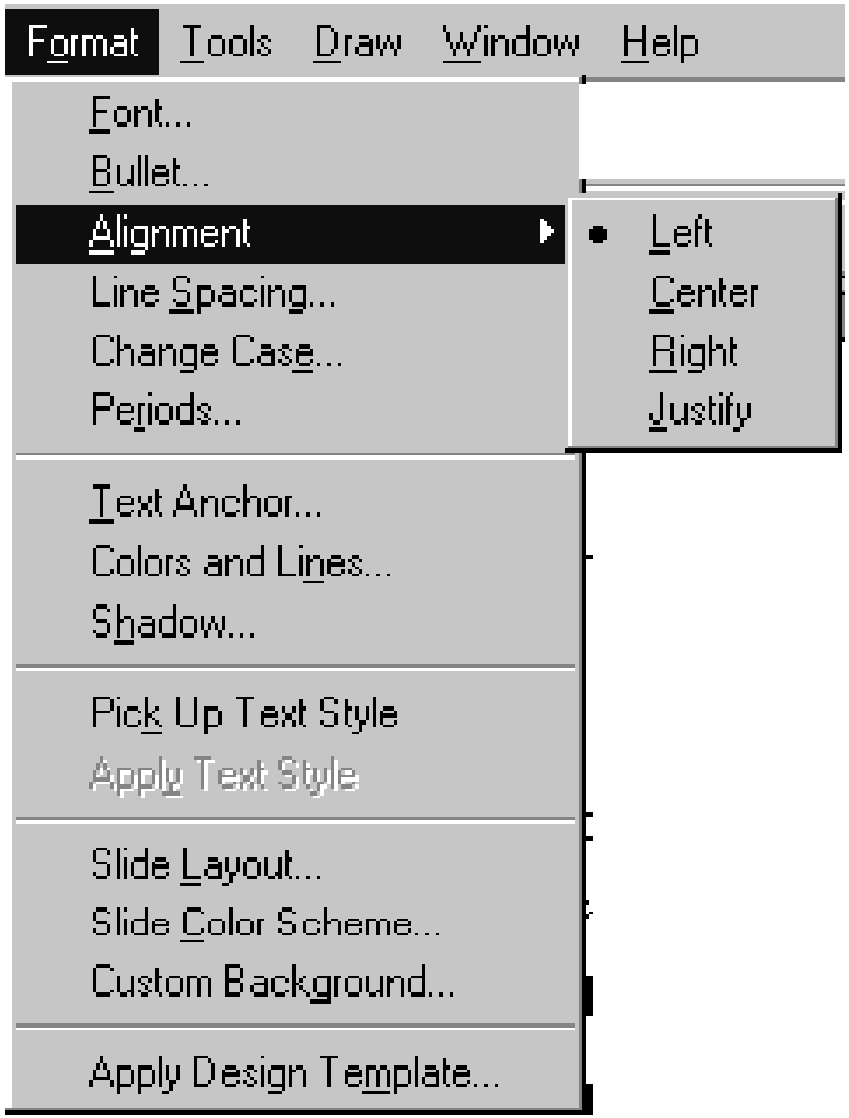
Priorities

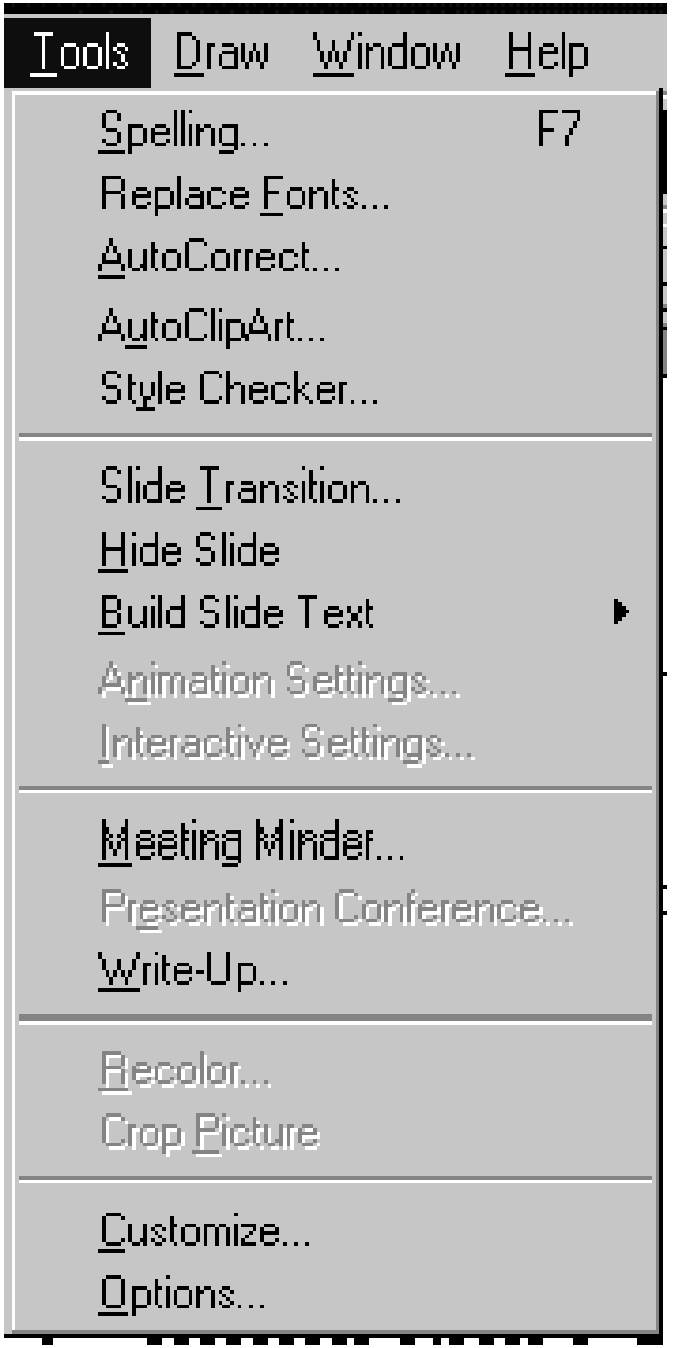
- Conventional categories (file, edit,…)
- Use of dividers to break menus into groups
- Logical groups of related actions (cut,copy,paste)

Sequencing options within groups

- **consistency** - use the same relative order of items where the group is presented again
- **importance** - place important items first in the group
- **conventional** order e.g days of the week
- **order of use** - e.g 'copy' precedes 'paste'
- **frequency** of use
  - if frequency of option is known, place frequent items first
- **alphabetical** order

- What ordering rules have been applied in the next slide?


Format
Tools
Draw
Window
Help
Font...
Bullet...
Alignment
Line Spacing...
Change Case...
Periods...
Left
Center
Right
Justify
Text Anchor...
Colors and Lines...
Shadow...
Pick Up Text Style
Apply Text Style
Slide Layout...
Slide Color Scheme...
Custom Background...
Apply Design Template...


Tools
Draw
Window
Help
Spelling... F7
Replace Fonts...
AutoCorrect...
AutoClipArt...
Style Checker...
Slide Transition...
Hide Slide
Build Slide Text
Animation Settings...
Interactive Settings...
Meeting Minder...
Presentation Conference...
Write-Up...
Recolor...
Crop Picture
Customize...
Options...

Formatting recommendations

- split strings more than 6 alphanumerics into smaller groups

(bad)	(good)
ABBA347686A2	ABBA 347686 A2
ABBA456388A3	ABBA 456388 A3

- identical data should be presented in the same way even if varitions in input format are tolerated

  30 11 95

  30 Nov 1995 -> 30/11/95 (for example)

  30 11 1995

  30th nov 95

Formatting recommendations

- data should be presented in full version even if abbreviated input allowed, provide feedback to user

Party:[ ]
Party:[ch,cai] Chemical Bank, Cairo

Formatting recommendations

- numeric codes displayed with right justification

47321	47321
539	539
67	67
482645	482645

- lists of numeric with decimal points should be aligned around the point

34.723
43.908
2341.5

Labeling in screen design

- descriptive title or phrase adjacent to a group of related items or information
- ensure labels are meaningful to the user
- labeling should be visually distinct from the data
- data labeling should not be able to be confused with help messages or command descriptions

Labeling in screen design

- use consistent relationship between labels and data being described

  e.g. above and left justified

  Name:

  [ ]

- include units in label to reduce ambiguity

  e.g. Weight(Kg):

  [ ]

Style guides and sources of design guidance

- Plenty of these….
- Manufacturers
- Web-based style guides e.g., Yale Style Manual
  - http://info.med.yale.edu/caim/manual/index.html