Chapter 11
User support

- Issues
  - different types of support at different times
  - implementation and presentation both important
  - all need careful design
- Types of user support
  - quick reference, task specific help, full explanation, tutorial
- Provided by help and documentation
  - help - problem-oriented and specific
  - documentation - system-oriented and general
  - same design principles apply to both

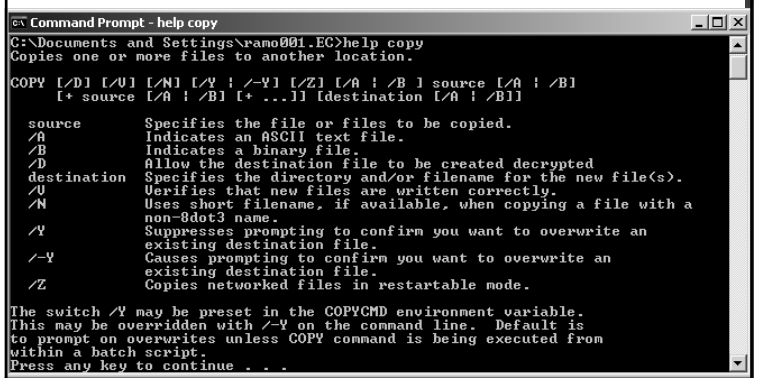
Requirements

- Availability
  - continuous access concurrent to main application
- Accuracy and completeness
  - help matches and covers actual system behaviour
- Consistency
  - between different parts of the help system and paper documentation
- Robustness
  - correct error handling and predictable behaviour
- Flexibility
  - allows user to interact in a way appropriate to experience and task
- Unobtrusiveness
  - does not prevent the user continuing with work

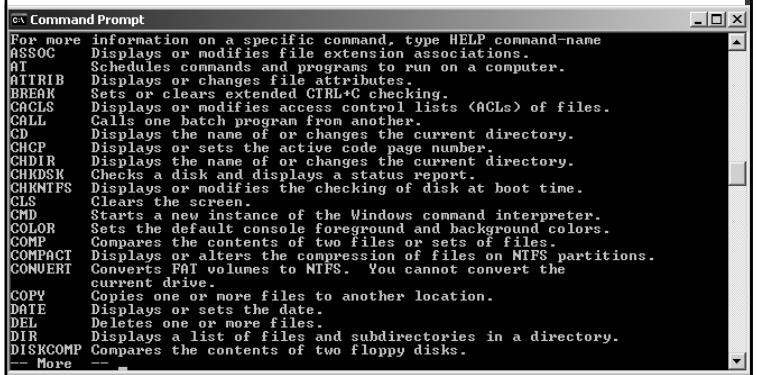
Approaches to user support

- Command assistance
  - User requests help on particular command
    e.g., UNIX man, DOS help
  - Good for quick reference
  - Assumes user know what to look for
- Command prompts
  - Provide information about correct usage when an error occurs
  - Good for simple syntactic errors
  - Also assumes knowledge of the command

Command assistance

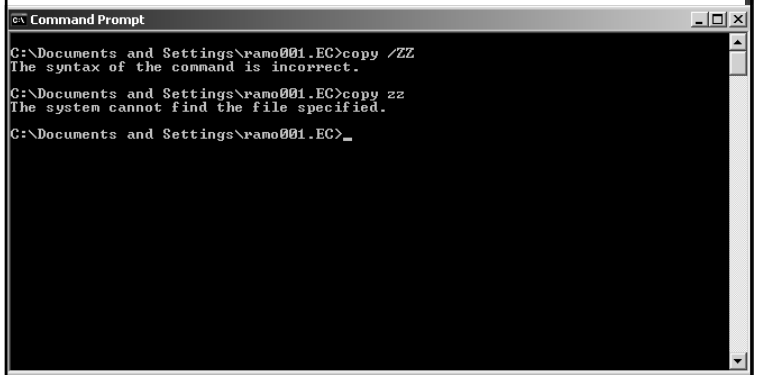


Finding a command

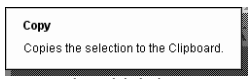


In unix we'd use "man –k" to find commands related to a keyword

Command prompts – not in DOS

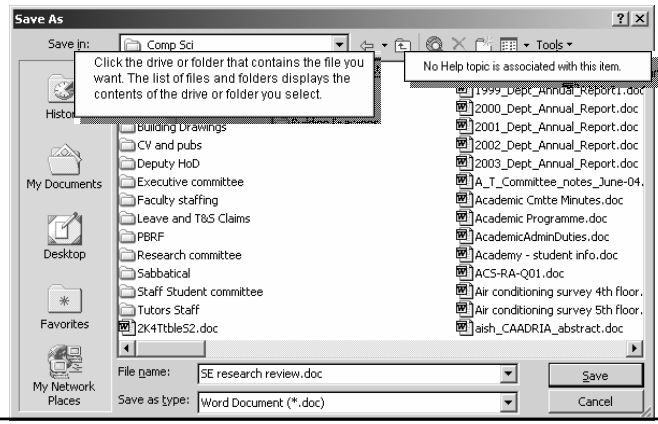


Approaches to user support (ctd)

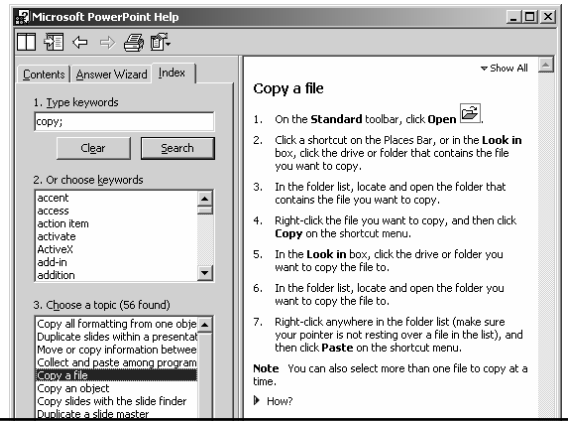


- Context sensitive help
  - help request interpreted according to context in which it occurs. e.g. tooltips
- On-line tutorials
  - user works through basics of application in a test environment.
  - can be useful but are often in flexible.
- On-line documentation
  - paper documentation is made available on computer.
  - continually available in common medium
  - can be difficult to browse
  - hypertext used to support browsing.

Context sensitive help



On-line documentation

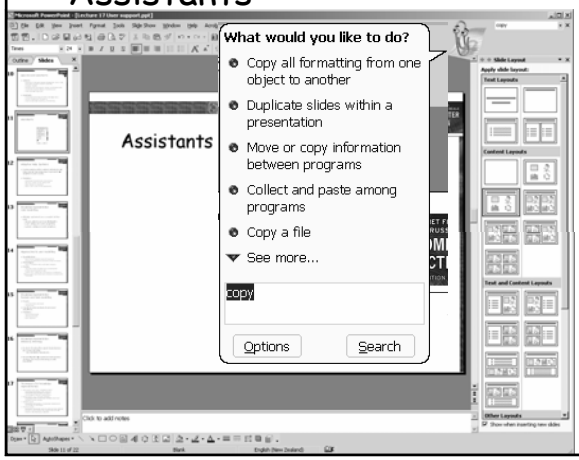


wizards and assistants

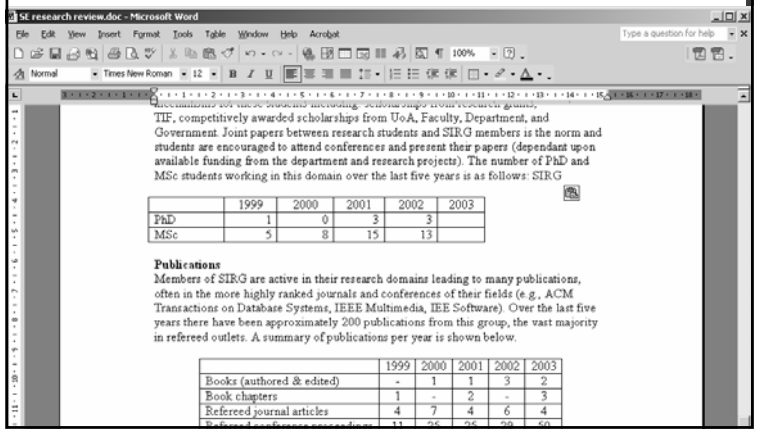
- wizards
  - task specific tool leads the user through task, step by step, using user's answers to specific questions
  - example: résumé
  - useful for safe completion of complex or infrequent tasks
  - constrained task execution so limited flexibility
  - must allow user to go back
- assistants
  - monitor user behaviour and offer contextual advice
  - can be irritating e.g. MS paperclip
  - must be under user control e.g. XP smart tags

Wizards

Assistants



Smart tags



Adaptive Help Systems

- Use knowledge of the context, individual user, task, domain and instruction to provide help adapted to user's needs.
- Problems
  - knowledge requirements considerable
  - who has control of the interaction?
  - what should be adapted?
  - what is the scope of the adaptation?

Knowledge representation
User modelling

- All help systems have a model of the user
  - single, generic user (non-intelligent)
  - user-configured model (adaptable)
  - system-configure model (adaptive)

Approaches to user modelling

- Quantification
  - user moves between levels of expertise
  - based on quantitative measure of what he knows.
- Stereotypes
  - user is classified into a particular category.
- Overlay
  - idealized model of expert use is constructed
  - actual use compared to ideal
  - model may contain the commonality or difference

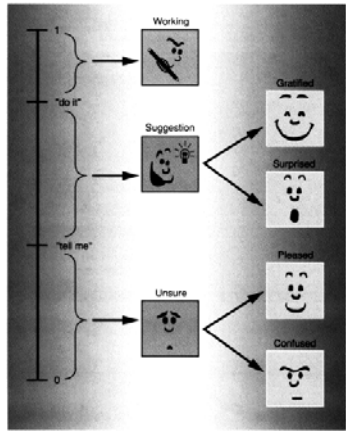
  Special case: user behaviour compared to known error catalogue

Knowledge representation
Domain and task modelling

- Covers
  - common errors and tasks
  - current task
- Usually involves analysis of command sequences.
- Problems
  - representing tasks
  - interleaved tasks
  - user intention

Anthropomorphic agents

- If we give agents a 'face' the metaphor is of an 'intelligent' assistant
  - Patti Maes espoused assistant agents in the '90s for sorting news, email, etc.
  - Combines probability with *agency*
  - An agent is something you can 'trust' to do a task for you
    - E.g., an e-commerce agent might make purchases or sales for you within specified parameters
    - It seems more like an agent and less like a tool when its reasoning is opaque



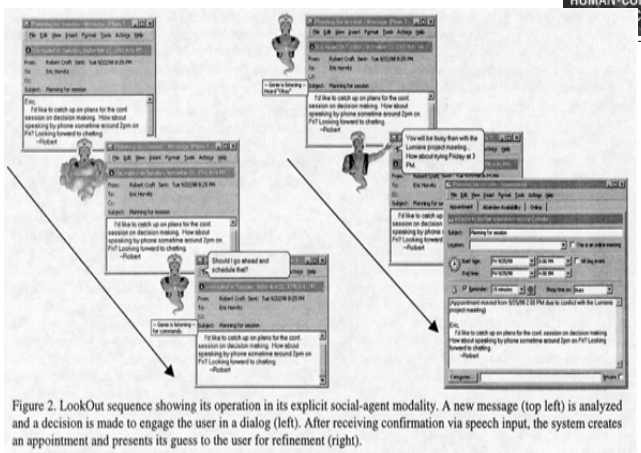


Figure 2. LookOut sequence showing its operation in its explicit social-agent modality. A new message (top left) is analyzed and a decision is made to engage the user in a dialog (left). After receiving confirmation via speech input, the system creates an appointment and presents its guess to the user for refinement (right).

http://www2.sims.berkeley.edu/courses/is296a-4/f99/papers/horvitz-chi99.pdf#search=%22lookout%20outlook%20horvitz%22

Including 'buggy rules'

- A great application of adaptive UI is in online learning environments
  - Also known as 'Intelligent Tutoring Systems'
  - Want to represent the 'syllabus' (what user should know)
  - And an overlay template for each user (how well they know each concept)
  - And possibly common 'bugs' or errors that users make
    - Ways they commonly get a program or procedure wrong
    - They you can recognize the bug and give special advice on how to avoid it

Knowledge representation
Advisory strategy

- involves choosing the correct style of advice for a given situation.
  e.g. reminder, tutorial, etc.
- few intelligent help systems model advisory strategy, but choice of strategy is still important.

Issues in adaptive help

- Initiative
  - does the user retain control or can the system direct the interaction?
  - can the system interrupt the user to offer help?
- Effect
  - what is going to be adapted and what information is needed to do this?
  - only model what is needed.
- Scope
  - is modelling at application or system level?
  - latter more complex
    e.g. expertise varies between applications.

Designing user support

- User support is not an `add on'
  - should be designed integrally with the system
  - Common problem is that user support gets squeezed out as a project runs over time (bad mistake!)
- Concentrate on content and context of help rather than technological issues

Presentation issues

- How is help requested?
  - command, button, function (on/off), separate application
- How is help displayed?
  - new window, whole screen, split screen,
  - pop-up boxes, hint icons
- Effective presentation requires
  - clear, familiar, consistent language
  - instructional rather than descriptive language
  - avoidance of blocks of text
  - clear indication of summary and example information

Implementation issues

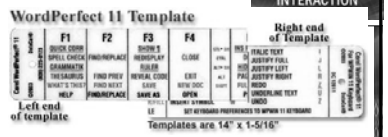
Is help
- operating system command
- meta command
- application

Structure of help data
- single file (XLM?)
- file hierarchy
- database

What resources are available?
- screen space (problem with online help is that it occupies the same screen as the application!)
- memory capacity
- speed

Issues
- flexibility and extensibility
- hard copy
- browsing

Design to user needs



- The User Support plan must fit the users' needs
  - Possibly multiple strategies for multiple types of users
  - Must fit the flow of work that you expect from the user
    - Will they have time for online help when they actually have a problem (in air traffic control)?
    - Can they reasonably be expected to do a tutorial or training course in advance?
    - Can we design a keyboard overlap template or quick reference card (Word Perfect had a great overlay)