Overview

Users require different types of support at different times but all user support should fulfill some basic requirements.

Implementation and presentation both need to be considered in designing user support.

Types of user support

- quick reference
- task specific help
- full explanation
- tutorial

These may be provided by help and/or documentation

- help — problem-oriented and specific
- documentation — system-oriented and general

The same design principles apply to both.

Requirements

Availability — continuous access concurrent to main application.

Accuracy and completeness — help matches actual system behaviour and covers all aspects of system behaviour.

Consistency — different parts of the help system and any paper documentation are consistent in content, terminology and presentation.

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 nor interfere with application.

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.

Context sensitive help

Help request interpreted according to context in which it occurs.
e.g., Spy, Balloons Help.

On-line tutorials

User works through basics of application in a test environment.
Can be useful but are often inflexible.

On-line documentation

Paper documentation is made available on computer.
Continually available in common medium but can be difficult to browse.
Hypertext used to support browsing.

Intelligent Help

Use knowledge of the 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

- quantification — user moves between levels of expertise based on quantitative measure of what he knows.
- stereotypes — user is classified into a particular category.
- overlay — an idealized model of expert use is constructed and actual use compared to it. Model may contain the commonality between these two or the difference. Special case: user behaviour compared to known error catalogue.

Domain and task modelling

Covers

- common errors and tasks
- current task

Usually involves analysis of command sequences.

Problems

- representing tasks
- interleaved tasks
- user intention

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.

Techniques for knowledge representation

- rule based — knowledge presented as rules and facts interpreted using inference mechanism. E.g. logic, production rules. Can be used in relatively large domains.

- frame based — knowledge stored in structures, each having slots which can be filled. Useful for a small domain.

- network based — knowledge represented as relationships between facts. E.g. semantic network. Can be used to link frames.

- example based — knowledge represented implicitly within decision structure. Trained to classify rather than programmed with rules. Requires little knowledge acquisition.

Problems with intelligent help systems

- knowledge acquisition
- resources
- interpretation of user behaviour

Issues in intelligent help systems

- 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’ — it should be designed integrally with the system.

Should 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

What resources are available?

- screen space
- memory capacity
- speed

Structure of help data

- single file
- file hierarchy
- database

Considerations

- flexibility and extensibility
- hard copy
- browsing