Guidelines, Principles and Theories

Guidelines

- Best practices
- Critics say:
  - Too specific, incomplete, hard to apply, and sometimes wrong
- Proponents say:
  - Encapsulate experience

Navigating the Interface

Sample of the National Cancer Institutes guidelines:

- Standardize task sequences
- Ensure that embedded links are descriptive
- Use unique and descriptive headings
- Use check boxes for binary choices
- Develop pages that will print properly
- Use thumbnail images to preview larger images

Some Accessibility Guidelines

- Provide a text equivalent for every non-text element
- Information conveyed with color should also be conveyed without it
- Title each frame to facilitate from identification and navigation

Example of Guidelines: Organizing the Display

Smith and Mosier (1986) offer five high-level goals

- Consistency of data display
- Efficient information assimilation by the user
- Minimal memory load on the user
- Compatibility of data display with data entry
- Flexibility for user control of data display

Example of Guidelines: Getting the User's Attention

Techniques for attracting attention:

- Intensity
- Marking
- Size
- Choice of fonts
- Inverse video
- Blinking
- Color
- Audio

Example of Guidelines: Facilitating data entry

Smith and Mosier (1986) offer five high-level objectives as part of their **guidelines** for data entry

- Consistency of data-entry transactions
- Minimal input actions by user
- Minimal memory load on users
- Compatibility of data entry with data display
- Flexibility for user control of data entry

Examples (1)

GUIDELINES FOR DESIGNING USER INTERFACE SOFTWARE

- Prepared for Deputy Commander for Development Plans and Support Systems, Electronic Systems Division, AFSC, United States Air Force, Hanscom Air Force Base, Massachusetts

Data Entry:

Vehicle Type? _

Vehicle Type? _	
C	Car
T	Truck
B	Bus

- Data Validation
- Error Prevention

Examples (2)

Data Entry

Datecode: _
SSN: _

Datecode: _ _ _ _ _ _
SSN: _ _ _ _ _ _ _ _ _ _ _
SSN (9 digits): _

Date: Week: _ Month: _ _ Year: _ _
Social Security No: _ _ _ - _ _ - _ _ _ _

- Informative Labels
- Field Length

SPEED LIMIT: _ _ miles per hour
FUEL USE: _ _ gallons per minute

BATAS KECEPATAN: _ _ km per jam
PENGGUNAAN BAHAN BAKAR: _ _ liter per km

- Unit yang familiar

Data Display

Update of Record A Record B Maintenance Change in Record C

Record A Change Record B Change Record C Change

Starting date: When did you quit: Home phone: Phone number at work:

Starting date: Leaving date: Home phone: Work phone:

- Consistent wording
- Consistent grammatical structure

Examples (3)

- Clear the screen before entering data. ✔
- Do not enter data before clearing the screen.

Clear the screen by pressing RESET. ✔
The screen is cleared by pressing RESET.

Enter LOGON before running programs. ✔
Before running programs enter LOGON.

- Affirmative
- Active voice

Examples (4)

Automobile Owners

Sara Alwine	2438	MA	929	448	103
Christopher Aranyi	2716	MA	797	AND	97
Maria Ashley	3397	MA	375	NRC	34
Arlene Atchison	7672	NH	60731		28
Steven Bahr	3272	MA	635	203	35
David Baldwin	3295	NH	63577		75
David Benkley	3581	MA	589	ADE	58
Marlin Boudreau	3413	MA	816	HER	93
Roger Cooksey	2144	MA	328	647	64
Joseph Curran	3167	RI	4693		83
Kent Delacy	3619	MA	749	827	29
Susan Doucette	2797	MA	525	115	41
Joseph Drury	7604	NH	42265		27
William Duvet	3898	MA	135	449	81
Samuel Everett	3619	MA	635	ASK	29
Jeanne Fiske	7642	MA	614	CSU	10
Nancy Graham	2358	MA	745	CKJ	10
Paul Greenbaum	3979	MA	846	BLN	103
Christopher Hesen	7544	MA	342	NCG	21
Joseph Hilsmith	2443	MA	286	PAM	100

- Logical organization
- Tables referenced by first column
- Row and column labels
- Consistent column spacing
- Column scanning cues
- Row scanning cues

Examples (5)

AUTOMOBILE OWNERS Page 1 of 4

LICENSE	EMPLOYEE	EXT	DEPT
MA 127 355	Michaels, Allison	7715	91
MA 135 449	Duvet, William	3898	81
MA 227 379	Smithson, Jill	2491	63
MA 227 GBH	Zadrowski, Susan	2687	53
MA 253 198	Jenskin, Erik	3687	31
MA 286 PAM	Hilsmith, Joseph	2443	100
MA 291 302	Leonard, John	7410	92
MA 297 210	Toth, Kelley	7538	45
MA 328 647	Cooksey, Roger	2144	64
MA 342 NCG	Hesen, Christopher	7544	21
MA 367 923	Maddox, Patrick	7070	66
MA 375 NRC	Ashley, Maria	3397	34
MA 376 386	Wheetley, Katherine	2638	58
MA 385 248	Malone, Frank	2144	64
MA 391 293	Lowman, Edward	8263	77

n = Next page

Examples (6)

Temperature at inlet valve (0F): _

- a = 300-400
- h = more than 400
- l = less than 300

Temperature at inlet valve (0F): _

- l = less than 300
- a = 300-400
- h = higher than 400

• Urutan Logis

Examples (7)

Cost		Output	
Actual	Predicted	Predicted	Actual
947	901	82	83
721	777	54	57
475	471	95	91

Cost		Output	
Actual	Predicted	Actual	Predicted
947	901	83	82
721	777	57	54
475	471	91	95

- Grouping for Data Comparison

Principles

- More fundamental, widely applicable, and enduring than guidelines
- Need more clarification
- Fundamental principles
  - Determine user’s skill levels
  - Identify the tasks
- Interaction styles (Ragam Dialog)
- Eight golden rules of interface design
- Prevent errors
- Automation and human control

Determine user's skill levels

- "Know thy user" Hansen (1971)
- Age, gender, physical and cognitive abilities, education, cultural or ethnic background, training, motivation, goals and personality
- Design goals based on skill level
  - Novice or first-time users
  - Knowledgeable intermittent users
  - Expert frequent users
- Multi-layer designs
  - Accommodate both novices and experts

Identify the tasks

- Task Analysis usually involve long hours observing and interviewing users
- Decomposition of high level tasks
- Relative task frequencies

	TASK				
Job title	Query by Patient	Update Data	Query across Patients	Add Relations	Evaluate System
Nurse	0.14	0.11			
Physician	0.06	0.04			
Supervisor	0.01	0.01	0.04		
Appointment personnel	0.26				
Medical-record maintainer	0.07	0.04	0.04	0.01	
Clinical researcher			0.08		
Database programmer			0.02	0.02	0.05

Choose an interaction style

- Direct Manipulation
- Menu selection
- Form filling
- Command language
- Natural language

Advantages	Disadvantages
Direct manipulation	
Visually presents task concepts	May be hard to program
Allows easy learning	May require graphics display and pointing devices
Allows easy retention	
Allows errors to be avoided	
Encourages exploration	
Affords high subjective satisfaction	
Menu selection	
Shortens learning	Presents danger of many menus
Reduces keystrokes	May slow frequent users
Structures decision making	Consumes screen space
Permits use of dialog-management tools	Requires rapid display rate
Allows easy support of error handling	
Form fillin	
Simplifies data entry	Consumes screen space
Requires modest training	
Gives convenient assistance	
Permits use of form-management tools	
Command language	
Is flexible	Has poor error handling
Appeals to "power" users	Requires substantial training and memorization
Supports user initiative	
Allows convenient creation of user-defined macros	
Natural language	
Relieves burden of learning syntax	Requires clarification dialog
	May not show context
	May require more keystrokes
	Is unpredictable

The 8 golden rules of interface design

1. Strive for consistency
2. Cater to universal usability
3. Offer informative feedback
4. Design dialogs to yield closure
5. Prevent errors
6. Permit easy reversal of actions
7. Support internal locus of control
8. Reduce short term memory

Prevent errors

- Make error messages specific, positive in tone, and constructive
- Mistakes and slips (Norman, 1983)
- Correct actions
  - Gray out inappropriate actions
  - Selection rather than freestyle typing
  - Automatic completion
- Complete sequences
  - Single abstract commands
  - Macros and subroutines

Automation and human control

- Successful integration:
  - Users can avoid:
    - Routine, tedious, and error prone tasks
  - Users can concentrate on:
    - Making critical decisions, coping with unexpected situations, and planning future actions

Automation and human control (cont.)

- Supervisory control needed to deal with real world open systems
  - E.g. air-traffic controllers with low frequency, but high consequences of failure
  - FAA: design should place the user in control and automate only to improve system performance, without reducing human involvement

Automation and human control (cont.)

- User modeling for adaptive interfaces
  - keeps track of user performance
  - adapts behavior to suit user's needs
  - allows for automatically adapting system
    - response time, length of messages, density of feedback, content of menus, order of menu items, type of feedback, content of help screens
  - can be problematic
    - system may make surprising changes
    - user must pause to see what has happened
    - user may not be able to
      - predict next change
      - interpret what has happened
      - restore system to previous state

Theories

- Beyond the specifics of guidelines
- Principles are used to develop theories
- Descriptions/explanatory or predictive
- Motor task, perceptual, or cognitive

SELESAI

Explanatory and predictive theories

- **Explanatory theories:**
  - Observing behavior
  - Describing activity
  - Conceiving of designs
  - Comparing high-level concepts of two designs
  - Training
- **Predictive theories:**
  - Enable designers to compare proposed designs for execution time or error rates

Perceptual, Cognitive, & Motor tasks

- **Perceptual or Cognitive subtasks theories**
  - Predicting reading times for free text, lists, or formatted displays
- **Motor-task performance times theories:**
  - Predicting keystroking or pointing times

Taxonomy (explanatory theory)

- Order on a complex set of phenomena
- Facilitate useful comparisons
- Organize a topic for newcomers
- Guide designers
- Indicate opportunities for novel products.

Conceptual, semantic, syntactic, and lexical model

- Foley and van Dam offer a four-level approach
  - *Conceptual level:*
    - User's mental model of the interactive system
  - *Semantic level:*
    - Describes the meanings conveyed by the user's command input and by the computer's output display
  - *Syntactic level:*
    - Defines how the units (words) that convey semantics are assembled into a complete sentence that instructs the computer to perform a certain task
  - *Lexical level:*
    - Deals with device dependencies and with the precise mechanisms by which a user specifies the syntax
- Approach is convenient for designers
  - Top-down nature is easy to explain
  - Matches the software architecture
  - Allows for useful modularity during design

Stages of action models

- **Norman's seven stages of action**
  1. **Forming the goal**
  2. **Forming the intention**
  3. **Specifying the action**
  4. **Executing the action**
  5. **Perceiving the system state**
  6. **Interpreting the system state**
  7. **Evaluating the outcome**
- **Norman's contributions**
  - **Context of cycles of action and evaluation.**
  - ***Gulf of execution:* Mismatch between the user's intentions and the allowable actions**
  - ***Gulf of evaluation:* Mismatch between the system's representation and the users' expectations**

Stages of action models (cont.)

- Four principles of good design
  - State and the action alternatives should be visible
  - Should be a good conceptual model with a consistent system image
  - Interface should include good mappings that reveal the relationships between stages
  - User should receive continuous feedback
- Four critical points where user failures can occur
  - Users can form an inadequate goal
  - Might not find the correct interface object because of an incomprehensible label or icon
  - May not know how to specify or execute a desired action
  - May receive inappropriate or misleading feedback

Consistency through grammars

Consistent user interface goal

- Definition is elusive - multiple levels sometimes in conflict
- Sometimes advantageous to be inconsistent.

Consistent	Inconsistent A	Inconsistent B
delete/insert character	delete/insert character	delete/insert character
delete/insert word	remove/bring word	remove/insert word
delete/insert line	destroy/create line	delete/insert line
delete/insert paragraph	kill/birth paragraph	delete/insert paragraph

Consistency through grammars (cont.)

Inconsistent action verbs

- Take longer to learn
- Cause more errors
- Slow down users
- Harder for users to remember

Consistency through grammars (cont.)

- Task-action grammars (TAGs) try to characterize a complete set of tasks.
- Example: TAG definition of cursor control

- Dictionary of tasks:

move-cursor-one-character-forward [Direction=forward,Unit=char]

move-cursor-one-character-backward [Direction=backward,Unit=char]

move-cursor-one-word-forward [Direction=forward,Unit=word]

move-cursor-one-word-backward [Direction=backward,Unit=word]

Consistency through grammars (cont.)

High-level rule schemas describing command syntax:

1. task [Direction, Unit] -> symbol [Direction] + letter [Unit]
2. symbol [Direction=forward] -> "CTRL"
3. symbol [Direction=backward] -> "ESC"
4. letter [Unit=word] -> "W"
5. letter [Unit=char] -> "C"

Generates a consistent grammar:

move cursor one character forward	CTRL-C
move cursor one character backward	ESC-C
move cursor one word forward	CTRL-W
move cursor one word backward	ESC-W

Widget-level theories

Follow simplifications made in higher-level, user-interface building tools.

Potential benefits:

- Possible automatic generation of performance prediction
- A measure of layout appropriateness available as development guide
- Estimates generated automatically and amortized over many designers and projects
    - perceptual complexity
    - cognitive complexity
    - motor load
- Higher-level patterns of usage appear

Object/Action Interface model

Syntactic-semantic model of human behavior

- used to describe
  - programming
  - database-manipulation facilities
  - direct manipulation
- Distinction made between meaningfully-acquired semantic concepts and rote-memorized syntactic details
- Semantic concepts of user's tasks well-organized and stable in memory
- Syntactic details of command languages arbitrary and required frequent rehearsal

Object/Action Interface model (cont.)

With introduction of GUIs, emphasis shifted to simple direct manipulations applied to visual representations of objects and actions.

Syntactic aspects not eliminated, but minimized.

Object/Action Interface model (cont.)

Object-action design:

2. understand the task.
   - real-world objects
   - actions applied to those object
3. create metaphoric representations of interface objects and actions
4. designer makes interface actions visible to users

Task hierarchies of objects and actions

Decomposition of real-world complex systems natural

- human body
- buildings
- cities
- symphonies
- baseball game

Task hierarchies of objects and actions (cont.)

Computer system designers must generate a hierarchy of objects and actions to model users' tasks:

- Representations in pixels on a screen
- Representations in physical devices
- Representations in voice or other audio cue

Interface hierarchies of objects and actions

Interface includes hierarchies of objects and actions at high and low levels. E.g. A computer system:

- Interface Objects
  - directory
    - name
    - length
    - date of creation
    - owner
    - access control
  - files of information
    - lines
    - fields
    - characters
    - fonts
    - pointers
    - binary numbers

Interface hierarchies of objects and actions (cont.)

- Interface Actions
  - load a text data file
  - insert into the data file
  - save the data file
    - save the file
    - save a backup of the file
    - apply access-control rights
    - overwrite previous version
    - assign a name

Interface hierarchies of objects and actions (cont.)

Interface objects and actions based on familiar examples.

Users learn interface objects and actions by:

- seeing a demonstration
- hearing an explanation of features
- conducting trial-and-error sessions

The disappearance of syntax

- Users must maintain a profusion of device-dependent details in their human memory.
  - Which action erases a character
  - Which action inserts a new line after the third line of a text file
  - Which abbreviations are permissible
  - Which of the numbered function keys produces the previous screen.

The disappearance of syntax (cont.)

- Learning, use, and retention of this knowledge is hampered by two problems
  - Details vary across systems in an unpredictable manner
  - Greatly reduces the effectiveness of paired-associate learning
- Syntactic knowledge
  - conveyed by example and repeated usage
  - often system dependent

The disappearance of syntax (cont.)

- Minimizing these burdens is the goal of most interface designers
  - Modern direct-manipulation systems
  - Familiar objects and actions representing their task objects and actions.
  - Modern user interface building tools
  - Standard widgets