Interaction Design

- Specifying the action sequences for planning and achieving one or more task goals
  - conveying what system goals are possible, plans for accomplishing them, physical actions to execute
- Usability engineering of an interaction design
  - ensure users can predict how to pursue goals, and that doing so is a comfortable and pleasant experience
  - (as usual) depends inherently on task, hence an important role of user interaction scenarios
1. Selecting a System Goal

- Going from users’ task concept to system concept: the cognitive distance between two models
  - user’s mental model guides decisions & activity
  - must be congruent with the designers’ model that is conveyed and supported by the user interface
- The closer the match, the easier to identify and achieve a relevant goal

Suggesting Goals to the User

- Label following: menu titles, folder names, application names, ...
- Decreasing the distance via direct manipulation:
  - UI controls appear as physical analogs of real objects; their affordances suggest interaction goals
  - key ideas are visual representation, immediate and continuing feedback, and simple reversibility
- Visual or auditory UI elements sometimes lead to opportunistic selection of goals
  - interesting object or message intrudes on a task
  - or user is paused, choosing among things to do; especially common among novice users
What are some *downsides* of direct manipulation?
Recall instead of Recognition

- Commands refer to system objects and actions
  - cognitive distance determined by words and phrases a system understands as user requests
  - recall more demanding, but flexible & saves screen space
  - design issues: vocabulary size and structure, familiarity and ambiguity (as discussed earlier)
  - also the syntax (grammar) of the command language
- Compromise: nested menus support hierarchy-path cued recall of a large set of commands
  - e.g., Format → Alignment → Center
- What about using natural language for commands?

Action Planning

- Plan analysis of required action sequences
  - like HTA, goals decomposed into subgoals, steps, etc.
  - includes choices and decision rules as relevant
  - examine what plan knowledge is expected of the user
  - look for arbitrary sequences, overall complexity, consistency, interference from one plan to another
- E.g.: action plan for changing to double-space?
Modeling Plan Knowledge with GOMS

- goals: unit task (delete word, create directory)
  - in general, verb-noun
- operators: elementary user actions
- methods: operator sequences to achieve a goal
- selection rules: choose among methods

Selection rule for goal of <general goal description>

IF <condition> THEN accomplish goal of <specific goal description>
IF <condition> THEN accomplish goal of <specific goal description>
...
Report goal accomplished

Example GOMS model for window closing

GOAL: CLOSE-ACTIVE-WINDOW
  [Select GOAL: USE-MENU-METHOD
    MOVE-MOUSE-TO-MENU-BAR
    DRAG-DOWN-FILEMENU
    RELEASE-ON-CLOSE-OPTION
    GOAL: USE-HANDLE-METHOD
    MOVE-MOUSE-TO-CORNER
    CLICK-ON-CLOSE-BOX
    GOAL: USE-CONTROL-KEY
    PRESS-APPLE+W ]

- What is a plausible selection rule?
Alternative methods for paragraph delete

- move cursor to top of paragraph (P)
- press button (K)
- move cursor to end of paragraph (P)
- release button (K)
- press DEL key (H+K)

- position cursor within paragraph (P)
- triple click (2-3K?)
- press DEL key (H+K)
- method done (M)

- method done (M)

- selection rule: if paragraph is long, then use method in right column
- prediction of task performance times

Designing Learnable Action Plans

- Try to make the sequence of actions match how user thinks about the real world task
  - design (or select from toolkit) interaction widgets that have good affordances
- Design a sequence, then analyze and refine
  - limited storage capacity of short-term memory (7 +/- 2)
  - look for ways to *chunk* long sequences of steps
  - use intermediate feedback or physical action to “mark” boundaries of subplans
  - ensure that subplan chunks match task subgoals
Guiding Action Planning with Forms/Dialo{

gs

- Users are already familiar with form fill-out
- Procedure (action plan) is implicit in the layout
  - numbering or instructions may emphasize
  - tabs, or auto-advance for convenience
- Design issues similar to those with menus
  - size and complexity can be intimidating
  - decomposing into simpler forms that match task;
    grouping subsets of fields or controls
  - flexibility, not everything is text-based
  - consistency in look and feel
Giving Control to the User

- Humans are good at—and expect support for—doing multiple things at once
  - working in parallel or “stacking” then reinitiating a task
  - internal rather than external locus of control
  - when computer drags them along, it’s unpleasant!
- BUT, what does this mean for interaction design?

Multi-threaded Interaction

- Multiple windows, each holds one thread
  - tradeoffs between tiled versus overlapping?
  - implies good support for window management
- Avoid modal dialogs unless they have task purpose
  - e.g., a preemptive dialog box that must be dismissed
  - when should you deliberately include modal dialog?
- Crucial role of status information
  - users must be able to tell when they return to a window
    what they have done so far, what is possible now, etc.
  - scenario-based design of “picking up the pieces”