Theories of User Interface Design
High-Level Theories

• Foley and van Dam four-level approach
• GOMS—Goals, Operators, Methods, and Selection Rules.
Foley and van Dam four-level approach

- Conceptual level:
  - User's mental model of the interactive system

As described by Jacobs, Tufts University
Foley and van Dam four-level approach

• Semantic level:
  – Describes the meanings conveyed by the user's command input and by the computer's output display
  – Documents the semantic specification for each action you have identified in the conceptual design, plus any other actions which are needed.
Foley and van Dam four-level approach

• 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
Foley and van Dam four-level approach

• Lexical level:
  – Deals with device dependencies and with the precise mechanisms by which a user specifies the syntax
Examples

Text editing
• **Conceptual: Provides a mental model**
  – text editor objects = characters, files, paragraphs
    relationships = files contain paragraphs contain chars
    operations = insert, delete, etc.

• **Semantic: meaning/desired function**
  – move the paragraph

• **Syntactic: how the semantic command is formed**
  – prefix vs. postfix
    (Edit, Highlight, Cut, Paste)

• **Lexical: sequence of actions**
  – how mouse and keyboard combined into menu, button, string, pick, etc.
  – Point to edit on menu bar->click ->select option within edit menu.
Foley and van Dam four-level approach

• Approach is convenient for designers
  – Top-down nature is easy to explain
  – Matches the software architecture
  – Allows for useful modularity during design

• Foundations for many other theories

• But, not used much today
  – It is more text-oriented
GOMS

Goals, Operators, Methods, and Selection Rules.

• descriptions of the Methods needed to accomplish specified Goals.
• Method: a series of Operators

The following slides are adapted from
http://www.cs.umd.edu/class/fall2002/cmsc838s/tichi/printer/goms.htm, and
http://bmrc.berkeley.edu/courseware/cs160/spring97/discussions/GOMSI
Members of GOMS Family

- Keystroke-Level Model (KLM) - Card, Moran, Newell (1983)
- CMN-GOMS (from TPoHCI)
- Natural GOMS Language (NGOMSL) - Kieras (1988+)
- Critical Path Method or Cognitive, Perceptual, and Motor GOMS (CPM-GOMS) - John (1990+)
What GOMS can model

- Task must be goal-directed
  - Some activities are more goal-directed than others
  - Even creative activities contain goal-directed tasks
- Task must a routine cognitive skill - as opposed to problem solving as in Cognitive Walkthrough
- Serial and parallel tasks
A Sample GOMS Model

- GOAL: EDIT-MANUSCRIPT
  - GOAL: EDIT-UNIT-TASK ... repeat until no more unit tasks
    - GOAL: ACQUIRE UNIT-TASK
      - GOAL: GET-NEXT-PAGE ... if at end of manuscript page
      - GOAL: GET-FROM-MANUSCRIPT
    - GOAL: EXECUTE-UNIT-TASK ... if a unit task was found
      - GOAL: MODIFY-TEXT
        - [select: GOAL: MOVE-TEXT* ... if text is to be moved
          - GOAL: DELETE-PHRA ... if a phrase is to be deleted
          - GOAL: INSERT-WORD] ... if a word is to be inserted
        - VERIFY-EDIT
      - [select**: GOAL: HIGHLIGHT-WORD
        - MOVE-CURSOR-TO-WORD
        - DOUBLE-CLICK-MOUSE-BUTTON
        - VERIFY-HIGHLIGHT
        - GOAL: HIGHLIGHT-ARBITRARY-TEXT
          - MOVE-CURSOR-TO-BEGINNING 1.10
          - CLICK-MOUSE-BUTTON 0.20
          - MOVE-CURSOR-TO-END 1.10
          - SHIFT-CLICK-MOUSE-BUTTON 0.48
          - VERIFY-HIGHLIGHT]
          - MOVE-CURSOR-TO-EDIT-MENU 1.10
          - PRESS-MOUSE-BUTTON 0.10
          - MOVE-CURSOR-TO-CUT-ITEM 1.10
          - VERIFY-HIGHLIGHT 1.35
          - RELEASE-MOUSE-BUTTON 0.10
    - GOAL: ISSUE-CUT-COMMAND
      - MOVE-CURSOR-TO-EDIT-MENU 1.10
      - PRESS-MOUSE-BUTTON 0.10
      - MOVE-CURSOR-TO-CUT-ITEM 1.10
      - VERIFY-HIGHLIGHT 1.35
      - RELEASE-MOUSE-BUTTON 0.10
  - GOAL: PASTE-TEXT
    - GOAL: POSITION-CURSOR-AT-INSERTION-POINT
      - MOVE-CURSOR-TO-INSERTION-POIONT 1.10
      - CLICK-MOUSE-BUTTON 0.20
      - VERIFY-POSITION 1.35
    - GOAL: ISSUE-PASTE-COMMAND
      - MOVE-CURSOR-TO-EDIT-MENU 1.10
      - PRESS-MOUSE-BUTTON 0.10
      - MOVE-MOUSE-TO-PASTE-ITEM 1.10
      - VERIFY-HIGHLIGHT 1.35
      - RELEASE-MOUSE-BUTTON 0.10

TOTAL TIME PREDICTED (SEC) 14.38
GOMS Output

• Functionality coverage and consistency
  – Does UI contain needed functions?
  – Are similar tasks performed similarly?

• Operator sequence
  – In what order are individual operations done?
  – Abstraction of operations may vary among models
• Execution time
  – By expert
  – Very good rank ordering
  – Absolute accuracy ~10-20%

• Procedure learning time (NGOMSL only)
  – Accurate for relative comparison only
  – Does not include time for learning domain knowledge

• Error recovery
Applications of GOMS analysis

• Compare UI designs
• Profiling
• Sensitivity and parametric analysis
• Building a help system
  – GOMS modeling makes user tasks and goals explicit
  – Can suggest questions users will ask and the answers
Keystroke-Level Model

• How to make a KLM
  – List specific actions user does to perform task
    • Keystrokes and button presses
    • Mouse movements
    • Hand movements between keyboard & mouse
    • System response time (if it makes user wait)
  – Add Mental operators
  – Assign execution times to steps
  – Sum execution times
• Only provides execution time and operator sequence
Natural GOMS Language (NGOMSL)

Natural-language notion for representing GOMS models

Method for goal: Cut text
   Step 1. Accomplish goal: Highlight text.
   Step 2. Return that the command is CUT, and accomplish goal: Issue a command.
   Step 3. Return with goal accomplished.

Selection rule set for goal: Highlight text
   If text-is word, then accomplish goal: Highlight word.
   If text-is arbitrary, then accomplish goal: Highlight arbitrary text.
   Return with goal accomplished.

Method for goal: Highlight arbitrary text
   Step 1. Determine position of beginning of text (1.20 sec)
   Step 2. Move cursor to beginning of text (1.10 sec)
   Step 3. Click mouse button. (0.20 sec)
Cognitive-Perceptual-Motor GOMS (CPM-GOMS)

- Assumes that operators can be performed in parallel
- Schedule chart (a.k.a. a PERT chart) to represent operators and dependencies between operators.
How to do (CMN-)GOMS Analysis

• Choose GOMS family member, making sure GOMS is appropriate for you

• Generate task description
  – Pick high-level user Goal
  – Write Method for accomplishing Goal - may invoke subgoals
  – Write Methods for subgoals
    • This is recursive
    • Stops when Operators are reached
How to do GOMS Analysis (cont’d)

• Evaluate description of task
• Apply results to UI
• Iterate
• Warning: Make sure you do not leave out actions!
  – Walkthrough can help
Operators vs. Methods

• Operator: the most primitive action
• Method: requires several Operators or subgoal invocations to accomplish
• Level of detail determined by
  – KLM level - keypress, mouse press
  – Higher level - select-Close-from-File-menu
  – Different parts of model can be at different levels of detail
Example 1: PDA Text Entry

- goal: enter-text-Newton
  - move-pen-to-text-start
  - goal: enter-word-Newton ...repeat until no more words
    - write-letter ...repeat until no more letters
    - [select: goal: correct-misrecognized-word] ...if incorrect

- expansion of correct-misrecognized-word goal
  - move-pen-to-incorrect-letter
  - write-letter
Example 2: Graph Drawer

- **goal: draw-graph**
  - **goal: draw-node**
    ...repeat until no more nodes
  - **goal: draw-circle**
    - **draw-circle-gesture**
    - **goal: verify-circle-gesture**
      » [select: goal: correct-gesture] ...if misrecognized or drawn incorrectly
  - **goal: connect-node**
    ...repeat until no more connections
    » **draw-line-gesture**
    » **move-pen-to-node-just-drawn**

- **goal: name-node**
  » **make-naming-gesture**
  » **goal:enter-text**
Example 2 (cont’d)

• expansion of correct-gesture goal
  – move-pen-to-undo-button
  – tap-undo-button

• goal: copy-node
  – move-pen-to-node
  – draw-copy-gesture
  – drag-pen-to-destination
Real-world GOMS Applications

- KLM
  - Mouse-based text editor
  - Mechanical CAD system
- NGOMSL
  - TV control system
  - Nuclear power plant operator’s associate
- CPM-GOMS
  - Telephone operator workstation
Advantages of GOMS

• Gives several qualitative and quantitative measures
• Model explains why the results are what they are
• Less work than user study
• Easy to modify when interface is revised
• Research ongoing for tools to aid modeling process
Disadvantages of GOMS

• Not as easy as heuristic analysis, guidelines, or cognitive walkthrough
• Only works for goal-directed tasks
• Assumes tasks are performed by expert users
• Evaluator must pick users’ tasks/goals
• Does not address several important UI issues, such as
  – readability of text
  – memorability of icons, commands
• Does not address social or organizational impact
Summary

• Provides info about many important UI properties
• Does not tell you everything you want to know about a UI
• Substantial effort to do initial model, but still easier than user testing
• Changing later is much less work than initial generation